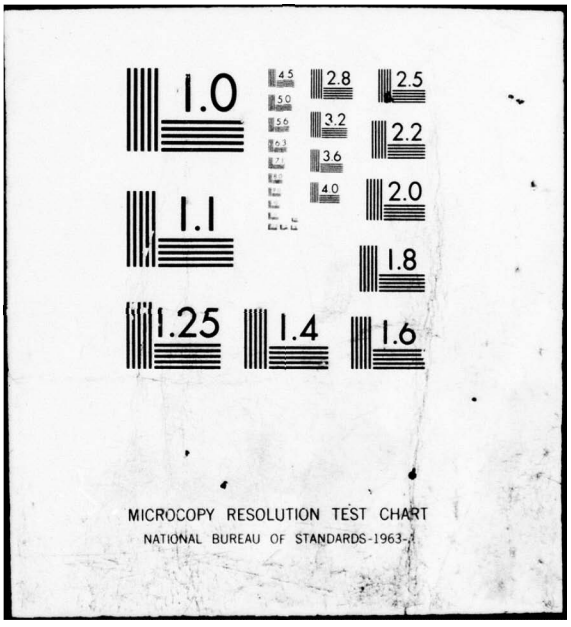
AD-A068 819 MASSACHUSETTS INST OF TECH CAMBRIDGE FLUID DYNAMICS --ETC F/G 20/4
AN OFF DESIGN SHOCK CAPTURING FINITE DIFFERENCE APPROACH FOR CA--ETC(U)
FEB 79 J R BARON, I EFRAT F49620-77-C-0090

UNCLASSIFIED AFOSR-TR-79-0517 NL

1 OF 1
AD A068819


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
5.6
6.3
7.1
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-

AFOSR-TR- 79-0517

Massachusetts Institute of Technology
Department of Aeronautics & Astronautics
Fluid Dynamics Research Laboratory

LEVEL II

AN OFF DESIGN SHOCK CAPTURING FINITE DIFFERENCE APPROACH FOR CARET WAVERIDER CONFIGURATIONS

J. R. BARON and I. EFRAT

FINAL TECHNICAL REPORT

AFOSR Contract F49620-77-C0090
M.I.T. OSP 85036

February 1979

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER: AFOSR-TR-79-0517	2. GOVT ACCESSION NO. / 3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle): An Off Design Shock Capturing Finite Difference Approach for Caret Waverider Configurations.	5. TYPE OF REPORT & PERIOD COVERED: Final Rept. 1 Apr 77 - 31 Dec 78
	6. PERFORMING ORG. REPORT NUMBER: MIT OSP-85036
7. AUTHOR(s): J. R. Baron and I. Efrat (Judean)	8. CONTRACT OR GRANT NUMBER(s): F49620-77-C-0090
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Massachusetts Institute of Technology, Department of Aeronautics & Astronautics, Cambridge, MA 02139	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 2307A1, 61102F, A1
11. CONTROLLING OFFICE NAME AND ADDRESS: AIR FORCE OFFICE OF SCIENTIFIC RESEARCH/NA, BLDG 410, BOLLING AIR FORCE BASE, D C 20332	12. REPORT DATE: February 1979
	13. NUMBER OF PAGES: 74
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) 95 P.	15. SECURITY CLASS. (of this report): UNCLASSIFIED
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Computational Fluid Dynamics
Waverider
Caret Wing
Shock Capturing
Finite Difference Method

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The three dimensional off design flow fields are calculated for stream Mach numbers in the range 1.3 to 4.0 and corresponding to attached and detached shocks at the leading edges of a reentrant pyramidal waverider geometry. The MacCormack shock capturing version of the Lax-Wendroff finite difference technique is used with grids chosen to align with surface, symmetry and approximate shock traces in the transverse plane. Separate natural grid systems are defined for the compression and expansion regions, and an

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED 140 250

20.

alternating region algorithm is used in combination with a sequential transfer of the edge region boundary conditions. The latter are derived from overlapping portions of the computational grids as integration proceeds axially to an asymptotic conical field. Equivalent attached shock cases result from either of two approaches: the alternating region algorithm, or a consideration of solely the compression region with uniform unknown conditions assumed near the edges. For detached shock cases overall lift and drag coefficients exhibit smooth variations between the attached edge and detached apex limits.

FOREWARD

This report summarizes work performed under United States Air Force sponsorship through the Office of Scientific Research, Contract F49620-77-C-0090. Contract monitor was Dr. James Wilson. Principal Investigator for M.I.T. was Professor Judson R. Baron.

ABSTRACT

Three dimensional off design flow fields are calculated for stream Mach numbers in the range 1.3 to 4.0 and corresponding to attached and detached shocks at the leading edges of a reentrant pyramidal waverider geometry. The MacCormack shock capturing version of the Lax-Wendroff finite difference technique is used with grids chosen to align with surface, symmetry, and approximate shock traces in the transverse plane. Separate natural grid systems are defined for the compression and expansion regions, and an alternating region algorithm is used in combination with a sequential transfer of the edge region boundary conditions. The latter are derived from overlapping portions of the computational grids as integration proceeds axially to an asymptotic conical field. Equivalent attached shock cases result from either of two approaches: the alternating region algorithm, or a consideration of solely the compression region with uniform unknown conditions assumed near the edges. For detached shock cases overall lift and drag coefficients exhibit smooth variations between the attached edge and detached apex limits.

CONTENTS

LIST OF FIGURES

SYMBOLS

a,b,c	boundary conditions parameters, eq.(4-29)
B	body surface, eq. (3-9)
C_D	drag coefficient
C_L	lift coefficient
C_p	pressure coefficient, $2(p-p_\infty)/\gamma M_\infty^2 p_\infty$
E,F,G,H	state property vectors, eq. (3-2)
$\bar{E},\bar{F},\bar{G},\bar{H}$	state property vectors, eq. (3-8)
E^*,F^*,G^*,H^*	state property vectors, eq. (4-11), (4-16)
g	surface geometry η dependence, eq. (3-27)
$\bar{G}$, G_s	eqs. (3-33), (3-35)
i,j	mesh indices for ψ,χ grid
i_m,j_m	maximum i, j
j_n	j^{th} grid line approximating normal to surface
j_{min}	eq. (4-18)
k	$(\gamma-1)/2\gamma$
k_1	eq. (4-31)
M	Mach number
n,j,k	mesh indices for state vectors E,F,G,H
$\bar{n},\bar{t}$	unit normal, tangent vectors
n_r,n_σ,n_θ	components of $\bar{n}$
p	pressure
P	eq. (3-26)
q	velocity magnitude
r, σ, θ	spherical coordinates
R	contents of radical, eq. (4-49)

S	shock surface, eq. (3-9)
s,s^*	dimensional, dimensionless semi-span, eq. (2-4)
t_r,t_σ,t_θ	components of $\bar{t}$
u,v,w	velocity components
u_c,v_c,w_c	velocity components in (r, σ, θ) system
v_{min},w_{min}	v,w for minimum (q^2-u^2), eq. (4-32)
x,y,z	Cartesian coordinates, Fig. 5
$\beta_{D,N,O,B}$	shock angle (on design, normal to side edge, off-design, baseplane)
β_i	$\tan^{-1}(\xi_i/\eta_i)$, eq. (4-44)
γ	specific heat ratio
δ, δ_N	wedge angle (symmetry plane, normal to side edge)
ζ,η,ξ	conical Cartesian coordinates, eq. (3-5)
ρ	density
τ	flow direction relative to edge, eq. (2-10)
ψ,χ	grid coordinates, eq. (4-6), (4-12)
ψ_u,ψ_ℓ	surface apex angles, eq. (2-2), (2-3)
$\bar{\psi},\bar{\chi}$	normalized grid coordinates, eq. (4-8)
$(\ )_2$	post shock condition
$(\ )_\infty$	free stream
$(\ )_B$	base(transverse) plane, Figure 1
$(\ )_C$	conical system, coordinate
$(\ )_D$	on-design

$( )_e$	upper (expansion) region
$( )_\ell$	lower (compression) region
$( )_N$	normal; control point for j_n grid line
$( )_o$	off-design
$( )_{opp}$	symmetrically opposite, eq. (4-43)
$( )_R$	ψ grid line origin, Figure 6
$( )_S$	shock; sonic
$( )_T$	tangential; edge tip
$( )_u$	χ grid line origin, Figure 6; upper surface
$( )_w$	wall
$( )'$	derivative with respect to η

1. INTRODUCTION

Waverider configurations have been of interest for some time for supersonic flight since the concept offers an implied control over confined pressure fields for practical three dimensional bodies. Conceptually an inverse approach is adopted to determine geometric surfaces consistent with fields derived from specified shock surfaces. Typically, but not necessarily, the shock surface is of a simple kind (planar or circular conic, e.g.), and the associated body corresponds to that specific (on-design) flow field. Nonweiler's original suggestion [1] was a delta planform, caret cross section wing of finite thickness and anhedral (Fig. 1) which matched a planar shock surface extending between the leading edges on the compression side. Gonor[2] and Maikapar[3] considered an integral number of similar components arranged circumferentially so as to form a star section, right reentrant pyramid. A number of studies have indicated that generalized waveriders under on-design conditions offer advantageous lift to drag ratios and drag reductions for equivalent volumes.[4,5]

The essential geometric features of the waverider cross section are the external and internal corners at the side edge and midspan (or reentrant rib) locations. The same features are common, of course, to edges and junctions of many geometric components of aeronautical interest and in particular for high speed engine inlets of a truncated pyramidal kind.

While on-design field evaluations are relatively simple, the off-design condition must be anticipated in most instances, and is in fact assured if viscous influences were to be taken into account. The shock surfaces are then unknown, rotational fields are certain to result, and shock detachment may occur. Several studies have explored the off-design field on the basis of integral methods, linear departures, and numerical methods, all for attached shock cases.[6-9] The detached leading edge case was considered on a corrected Newtonian theory basis [10] and free flight data for modified (blunt edge) caret sections is available.[11] The present objective is the application of a shock capturing finite difference method to caret waveriders for such off design conditions including detachment from the leading (side) edges. A second order, explicit algorithm is employed with grid systems defined explicitly for corner section configurations and approximate alignment with average shock trace locii. For the detached shock cases an alternating compression/ expansion region algorithm is used during the convergence process following the "time like" axial coordinate direction.

2. GEOMETRY, ON-DESIGN BASIS, OFF-DESIGN LIMITS

The caret wing geometry may be expressed in terms of the span, 2s, and the on-design flow deflection, δ, and shock, β_D, angles (Figure 1). For Mach numbers other than M_D the section angle normal to the leading edge controls shock attachment if the shock surface remains attached at the apex. The normal section angle is

$$\delta_N = \cos^{-1}\left\{\frac{\cos\beta_D \cos\delta \sin^2\psi_u - \sin\beta_D \sin\delta \cos^2\psi_u}{\sin\psi_u[\cos^2\beta_D - \cos^2\psi_u \cos^2(\beta_D-\delta)]^{1/2}}\right\} \tag{2.1}$$

where the upper surface apex angle is defined by

$$\psi_u = \cos^{-1}\left\{\frac{\cos\beta_D}{[1 + (s^*\cos\beta_D)^2]^{1/2}}\right\} \tag{2.2}$$

and similarly

$$\psi_\ell = \cos^{-1}\left\{\cos\psi_u \cos\delta(1 + \tan\beta_D \tan\delta)\right\} \tag{2.3}$$

Here s* is the local semispan in units of axial distance from the nose, or equivalently

$$s^* = \frac{\tan\beta_D}{\tan\beta_B} \tag{2.4}$$

in terms of the on-design shock angles in sections parallel and transverse to the mainstream.

The on-design condition is that for which identical disturbance fields result for oblique shocks based on either (M_D, δ) or (M_N, δ_N) pairs.

The normal section angle, δ_N, may be greater or less than δ depending upon the relative span (Figure 2), and for attached shocks the normal component of Mach number at the leading edge is

$$M_N = M_\infty \sin\psi_u = M_\infty \sin\left[\tan^{-1}(s^{*2} + \tan^2\beta_D)^{1/2}\right] \tag{2.5}$$

Thus the geometry defines a wide class of swept wing flow field behavior in the region "adjacent" to the leading edge if subject to attached oblique shock conditions. Typical parametric variations for geometries based on M_D = 2 and 4 are listed in Table 1. Figure 3 illustrates the span influence on the presence of weak or strong shocks at the side edge for the Table 1 cases applied to selected M_D and β_D levels.

A useful bound to the disturbance region when shocks are attached is provided by reinterpreting the geometry of eq. (2.1). With δ_N and δ replaced by β_N and β_o, i.e. by the shock angle normal to the leading edge and the angle (Figure 1) formed by the upper ridge line and the intersection of the shock surface with the symmetry plane,

$$\cos\beta_N = \frac{\cos\beta_D \cos\beta_o \sin^2\psi_u - \sin\beta_D \sin\beta_o \cos^2\psi_u}{\sin\psi_u \left[\cos^2\beta_D - \cos^2\psi_u \cos^2(\beta_D-\beta_o)\right]^{1/2}} \tag{2.6}$$

This is an implicit relation for the farthest possible extent of that shock surface; i.e.

$$\beta_o = \beta_o(\beta_N;\ \beta_D, s^*) \tag{2.7}$$

The on-design value of β_N follows from eq. (2.6) on taking $\beta_o = \beta_D$; there results

$$\cos(\beta_N)_D = \left\{\frac{s^*}{s^* + \tan^2\beta_D}\right\}^{1/2} \cos\beta_D \tag{2.8}$$

Typical values for β_N and β_o with departure of M_∞ from the design value are presented in Tables 2 and 3.

The design condition, geometry and stream Mach number, also can be viewed as the configuration for which the disturbance is imparted no lateral momentum. The flow downstream of the shock surface proceeds parallel to the geometric symmetry plane. A measure of off-design is therefore the turning relative to that plane, since the symmetry condition then implies that some corresponding compression or expansion must appear in order to redirect the flow. The flow direction, τ, from the leading edge and in the plane of the lower (compression)surface follows from the component Mach numbers after the attached shock (Figure 4)

$$M_{N_2}^2 = \frac{1 + \frac{\gamma-1}{2}(M_N\sin\beta_N)^2}{\sin^2(\beta_N-\delta_N)\left[\gamma(M_N\sin\beta_N)^2 - \frac{\gamma-1}{2}\right]}$$

$$M_{T_2}^2 = \frac{(M_\infty\cos\psi_u)^2}{1 + \frac{2(\gamma-1)}{(\gamma+1)^2}\frac{(M_N\sin\beta_N)^2-1}{(M_N\sin\beta_N)^2}\left[\gamma(M_N\sin\beta_N)^2 + 1\right]} \tag{2.9}$$

I.e.,

$$\tan\tau = \frac{M_{N_2}}{M_{T_2}} = \left\{ \frac{(M_N \sin\beta_N)^2 + 5}{6(M_N \sin\beta_N)\sin(\beta_N - \delta_N) M_\infty \cos\psi_u} \quad \text{if} \quad = \frac{7}{5} \right\} \qquad (2.10)$$

On-design this reduces to $\tau_D = \psi_\ell$, and $(\tau_D - \tau) \gtrless 0$ implies flow towards either the side edge or the symmetry plane. Table 3 includes some typical turning levels and the corresponding lateral component of velocity.

3. DESCRIPTIVE FIELD EQUATIONS AND BOUNDARY CONDITIONS

Conservation Equations

The governing equations for inviscid, steady flow, of a perfect gas, with pressure and density in units of undisturbed stream stagnation values and velocity components in units of the maximum adiabatic velocity, expressed in Cartesian coordinates are

$$E_x + F_y + G_z = 0 \tag{3.1}$$

where

$$E = \begin{bmatrix} \rho u \\ kp + \rho u^2 \\ \rho uv \\ \rho uw \end{bmatrix}, \quad F = \begin{bmatrix} \rho u \\ \rho uv \\ kp + \rho v^2 \\ \rho vw \end{bmatrix}, \quad G = \begin{bmatrix} \rho w \\ \rho uw \\ \rho vw \\ kp + \rho w^2 \end{bmatrix} \tag{3.2}$$

and $k = (\gamma-1)/2\gamma$. The adiabatic energy equation then specifies

$$p = \rho(1 - q^2) \tag{3.4}$$

For attached nose shocks an axial scale length is absent and a conical coordinate system is appropriate. If (Figure 5)

$$\begin{aligned} \zeta &= \ell n x \\ \eta &= \frac{y}{s} \quad \left(= \frac{y}{xs^*} \right) \\ \xi &= \frac{z}{s} \quad \left(= \frac{z}{xs^*} \right) \end{aligned} \tag{3.5}$$

then the (x,y,z) to (ζ,η,ξ) transformation implies

$$\frac{\partial}{\partial x} = \frac{1}{x}\frac{\partial}{\partial \zeta} - \frac{\eta}{x}\frac{\partial}{\partial \eta} - \frac{\xi}{x}\frac{\partial}{\partial \xi}$$

$$\frac{\partial}{\partial y} = \frac{1}{xs^*}\frac{\partial}{\partial \eta} \tag{3.6}$$

$$\frac{\partial}{\partial z} = \frac{1}{xs^*}\frac{\partial}{\partial \xi}$$

and so eqs,(3.1) become the weak conservation form

$$\bar{E}_\zeta + \bar{F}_\eta + \bar{G}_\xi + \bar{H} = 0 \tag{3.7}$$

with

$$\bar{E} = E \quad , \quad \bar{F} = \frac{F}{s^*} - \eta E$$

$$\bar{H} = 2E \quad , \quad \bar{G} = \frac{G}{s^*} - \xi E \tag{3.8}$$

Boundary Conditions

Say the body and shock surfaces are given by

$$B(r, \sigma, \theta) = 0 \quad , \quad S(r, \sigma, \theta) = 0 \tag{3.9}$$

in a spherical (r, σ, θ) coordinate system. A conical body geometry may then be taken in the form

$$B_c(\sigma, \theta) = \sigma - f(\theta) = 0 \tag{3.10}$$

with a unit surface normal $\bar{n} = (n_r, n_\sigma, n_\theta)$ given by

$$\bar{n} = \frac{\nabla B}{|\nabla B|} = \frac{(\frac{\partial B}{\partial r})\bar{i} + (\frac{1}{r}\frac{\partial B}{\partial \sigma})\bar{j} + (\frac{1}{r\sin\sigma}\frac{\partial B}{\partial \theta})\bar{k}}{[(\frac{\partial B}{\partial r})^2 + (\frac{1}{r}\frac{\partial B}{\partial \sigma})^2 + (\frac{1}{r\sin\sigma}\frac{\partial B}{\partial \theta})^2]^{1/2}}$$

$$= \frac{(\sin\sigma)\bar{j} - (f')\bar{k}}{(\sin^2\sigma + f'^2)^{1/2}} \qquad (3.11)$$

The unit surface tangent $\bar{t} = (t_r, t_\sigma, t_\theta)$, orthogonal to $\bar{i}$ and $\bar{n}$ in a positive θ sense, is

$$\bar{t} = \frac{\bar{i} \times \bar{n}}{|\bar{i} \times \bar{n}|} = -n_\theta\bar{j} + n_\sigma\bar{k} = (0, -n_\theta, n_\sigma) \qquad (3.12)$$

The velocity vector $\bar{q} = (u_c, v_c, w_c)$ has components at the surface which are

$$\bar{q}_N = (\bar{q}\cdot\bar{n})\bar{n} = \frac{v_c\sin\sigma - w_c f'}{(\sin^2\sigma + f'^2)^{1/2}}\ \bar{n}$$

$$\bar{q}_T = u_c\bar{i} + (\bar{q}\cdot\bar{t})\bar{t} = u_c\bar{i} + \frac{v_c f' + w_c\sin\sigma}{(\sin^2\sigma + f'^2)^{1/2}}\ \bar{t} \qquad (3.13)$$

and for $\bar{q}_N = 0$ it follows that

$$\left(\frac{w_c}{v_c}\right)_w = \frac{\sin\sigma}{f'} \qquad (3.14)$$

For the case of a planar conical surface such as in Figure 1 the eq. (3.10) description implies

$$f(\theta) = \tan^{-1}\left[\frac{s^* \tan\delta \sin\beta_\beta}{\tan\beta_D \sin(\theta-\beta_\beta) + \tan\delta \cos\theta \sin\beta_\beta}\right] \tag{3.15}$$

and

$$f'(\theta) = \sin\sigma \cos\sigma\left[\frac{\tan\delta \sin\theta \sin\beta_\beta - \tan\beta_D \cos(\theta - \beta_\beta)}{\tan\delta \cos\theta \sin\beta_\beta + \tan\beta_D \sin(\theta - \beta_\beta)}\right] \tag{3.16}$$

$$= - \frac{\sin\sigma \cos\sigma}{\tan(\theta - \beta_\beta + \delta_\beta)}$$

Here $(\theta-\beta_\beta+\delta_\beta)$ is the angle in the transverse (y,z) plane between the surface and the θ = constant plane. Thus eq. (3-14) becomes

$$\left(\frac{w_c}{v_c}\right)_w = - \frac{\tan(\theta - \beta_\beta + \delta_\beta)}{\cos\sigma} \tag{3.17}$$

and implies that

$$(\theta - \beta_\beta + \delta_\beta) = \begin{cases} 0 \\ \frac{\pi}{2} \end{cases} \text{ corresponds to } \left.\begin{matrix} w_{c_w} \\ v_{c_w} \end{matrix}\right\} = 0 \text{ when } \left.\begin{matrix} v_{c_w} \\ w_{c_w} \end{matrix}\right\} \text{lie in the surface plane.} \tag{3.18}$$

The normal and tangent expressions eqs. (3.11), (3.12) also are valid for a conical shock

$$S_c(\sigma,\theta) = \sigma_s - f_s(\theta) = 0 \tag{3.19}$$

and in the conical system the components of the free stream velocity

$$\bar{q}_\infty = (q_\infty \cos\sigma)\bar{i} - (q_\infty \sin\sigma)\bar{j} \tag{3.20}$$

are

$$\bar{q}_{\infty_N} = -\frac{q_\infty \sin^2\sigma_s}{(\sin^2\sigma_s + f_s'^2)^{1/2}}\ \bar{n}_s \tag{3.21}$$

$$\bar{q}_{\infty_T} = (q_\infty \cos\sigma_s)\bar{i} - \frac{q_\infty f_s' \sin\sigma_s}{\sin^2\sigma_s + f_s'^2}(f_s'\,\bar{j} + \sin\sigma_s \bar{k})$$

The shock angle is

$$\beta(\theta) \equiv \sin^{-1}\frac{|\bar{q}_{\infty_N}|}{q_\infty} = \sin^{-1}\left\{\frac{\sin^2\sigma_s}{(\sin^2\sigma_s + f_s'^2)^{1/2}}\right\} \tag{3.22}$$

and the post shock velocity components follow from the usual shock relations and

$$\bar{q}_s = \bar{q}_{s_N} + \bar{q}_{\infty_T}$$

$$\bar{q}_{s_N} = -q_{s_N}\bar{n}_s \tag{3.23}$$

I.e.,

$$q_{s_N} = q_{\infty_N}\frac{(\gamma-1)(M_\infty \sin\beta)^2 + 2}{(\gamma+1)(M_\infty \sin\beta)^2} \tag{3.24}$$

and

$$u_s = q_\infty \cos\sigma_s \qquad = u_{\infty s}$$

$$v_s = -q_\infty \sin\sigma_s \left(\frac{\frac{\gamma-1}{\gamma+1}\sin^2\sigma_s + f_s'^2}{\sin^2\sigma_s + f_s'^2} + \frac{2}{(\gamma+1)(M_\infty \sin\sigma_s)^2} \right) = v_{\infty s} \left(1 - \frac{\frac{p_s}{p_\infty} - 1}{\gamma (M_\infty \sin\sigma_s)^2} \right) \tag{3.25}$$

$$w_s = \frac{2}{\gamma+1} q_\infty f_s' \left(\frac{1}{(M_\infty \sin\sigma_s)^2} - \frac{\sin^2\sigma_s}{\sin^2\sigma_s + f_s'^2} \right) = \frac{q_\infty (1 - \frac{p_s}{p_\infty})}{\gamma M_\infty^2} \left(\frac{1}{P} - \frac{1}{\sin^2\sigma_s} \right)^{1/2}$$

where

$$P = \frac{\gamma+1}{2\gamma M_\infty^2} \left(\frac{p_s}{p_\infty} + \frac{\gamma-1}{\gamma+1} \right) \tag{3.26}$$

Boundary Conditions for Cartesian Conical System

For the modified Cartesian, conical, system of eqs. (3.5) - (3.8):

$$B_c = \xi - g(\eta) = 0$$

$$\bar{n} = \frac{s^*(\eta g' - \xi)\bar{i} - g'\bar{j} + \bar{k}}{([s^*(\eta g' - \xi)]^2 + g'^2 + 1)^{1/2}} = (n_\zeta, n_\eta, n_\xi) \tag{3.27}$$

The surface boundary condition $\bar{q}_N = (\bar{q}\cdot\bar{n})\bar{n} = 0$ then implies

$$\left(\frac{v}{u}\right)_w = -\left(\frac{n_\zeta + \frac{w}{u} n_\xi}{n_\eta} \right)_w = \left(\frac{s^*(\eta g' - \xi) + \frac{w}{u}}{g'} \right)_w \tag{3.28}$$

Also the tangent component is the surface velocity vector

$$(\bar{q}_T)_w \equiv \bar{q}_w = u_w \left[\bar{i} - \frac{n_\zeta + (\frac{w}{u})_w n_\xi}{n_\eta} \bar{j} + (\frac{w}{u})_w \bar{k} \right] \tag{3.29}$$

Specializing to a planar conical surface as in Figure 1, eq. (3.27) is specifically (Figure 5)

$$\xi = \begin{cases} \left(\frac{\tan\beta_D - \tan\delta}{s^*} \right) \eta + \frac{\tan\delta}{s^*} & \text{[lower surface]} \\ (\tan\beta_\beta)\, \eta & \text{[upper surface]} \end{cases} \tag{3.30}$$

E.g., for the lower surface eqs. (3.28), (3.29) are:

$$(\frac{v}{u})_w = \frac{(\frac{w}{u})_w - \tan\delta}{g'_\ell} \tag{3.31}$$

$$(\bar{q}_T)_w = u_w \left[\bar{i} + \frac{(\frac{w}{u})_w - \tan\delta}{g'_\ell} \bar{j} + (\frac{w}{u})_w \bar{k} \right]$$

At a shock surface $\xi_s = g_s(\eta)$ the shock normal is of the same form as in (3.27). Now, however,

$$\bar{q}_\infty = u_\infty \bar{i} \tag{3.32}$$

and the normal and tangential components are

$$(\bar{q}_\infty)_N = \frac{u_\infty}{(1 + G_s)^{1/2}} \quad \frac{s^*(\eta g_s' - \xi_s)}{|s^*(\eta g_s' - \xi_s)|} \bar{n}_s$$

$$(\bar{q}_\infty)_T = \frac{u_\infty}{(1 + G_s)} \left[G_s \bar{i} + \frac{g_{s_j}' \bar{j} - \bar{k}}{s^*(\eta g_s' - \xi_s)} \right] \tag{3.33}$$

where $G_s = (1 + g_s'^2)/[s^*(\eta g_s' - \xi_s)]^2$ and the shock angle is

$$\beta(\eta) = \sin^{-1} \frac{1}{(1 + G_s)^{1/2}} \tag{3.34}$$

The post shock components are then

$$u_s = \left[\frac{G_s - \bar{G}}{1 + G_s} \right] u_\infty$$

$$v_s = - g_s' w_s \tag{3.35}$$

$$w_s = - \left[\frac{1 + \bar{G}}{s^*(\eta g_s' - \xi_s)(1 + G_s)} \right] u_\infty$$

with

$$\bar{G} = \frac{(\gamma-1)M_\infty^2 + 2(1 + G_s)}{(\gamma+1)M_\infty^2} \left[\frac{s^*(\eta g_s' - \xi_s)}{|s^*(\eta g_s' - \xi_s)|} \right]$$

4. FINITE DIFFERENCE NUMERICAL SCHEME

Predictor Corrector Forms

The solution of the descriptive system, eqs. (3.7) was carried out using a finite-difference technique with the computation advanced along the time-like axial, ζ, coordinate until steady (axially uniform) conical conditions were achieved. The specific predictor/corrector algorithm was the alternating direction, explicit difference scheme of second order accuracy as suggested by MacCormack [12].

If the differential description is (eqs. (3.7))

$$\bar{E}_{\zeta} = -\bar{F}_{\eta} - \bar{G}_{\xi} - \bar{R} \tag{4.1}$$

and (n, j, k) are the indices for grid lines in the (ζ, η, ξ) directions then the predictor provides the intermediate state vector

$$\hat{\bar{E}}^{n+1}_{j,k} = \bar{E}^{n}_{j,k} - \frac{\Delta\zeta}{\Delta\eta}(\bar{F}^{n}_{j+1,k} - \bar{F}^{n}_{j,k}) - \frac{\Delta\zeta}{\Delta\xi}(\bar{G}^{n}_{j,k+1} - \bar{G}^{n}_{j,k}) - \bar{R}^{n}_{j,k}\,\Delta\zeta \tag{4.2}$$

and the corrector provides

$$\bar{E}^{n+1}_{j,k} = \frac{1}{2}\left[\bar{E}^{n}_{j,k} + \hat{\bar{E}}^{n+1}_{j,k} - \frac{\Delta\zeta}{\Delta\eta}(\hat{\bar{F}}^{n+1}_{j,k} - \hat{\bar{F}}^{n+1}_{j-1,k}) - \frac{\Delta\zeta}{\Delta\eta}(\hat{\bar{G}}^{n+1}_{j,k} - \hat{\bar{G}}^{n+1}_{j,k-1}) - \hat{\bar{R}}^{n+1}_{j,k}\,\Delta\zeta\right] \tag{4.3}$$

Here

$$\bar{E}^{n}_{j,k} = \bar{E}(n\Delta\zeta, j\Delta\eta, k\,\Delta\xi) \tag{4.4}$$

and from eqs. (3.2) and (3.8)

$$\begin{aligned} \bar{F}^n_{j,k} &= \bar{F}(E^n_{j,k}) \\ \bar{G}^n_{j.k} &= \bar{G}(E^n_{j,k}) \\ \bar{H}^n_{j.k} &= \bar{H}(E^n_{j.k}) \end{aligned} \tag{4.5}$$

A solution is converged upon achieving constant $\bar{E}^n_{j,k}$ after advancing n a sufficient number of times.

Grid Systems

For planar surface waveriders the natural grid system is one that conforms to the surface, symmetry, and shock traces in the (η,ξ) plane. The shock-capturing capability of the MacCormack algorithm and the unknown location of the jump distribution suggests the quadrilateral (APBC) bounded grid formed by new coordinates (χ,ψ) which are defined as (Figure 6)

$$\begin{aligned} \chi &= \frac{\eta - \eta_u}{\xi_u - \xi} \\ \psi &= \frac{\xi - \xi_R}{\eta_R - \eta} \end{aligned} \tag{4.6}$$

These are essentially angular measures from the surface and symmetry planes for the lower (compression) side. Approximate grid alignment with shock boundaries then follows from choices for the (η_R,ξ_R) and (η_u, ξ_u) centers and (χ_{max}, ψ_{max}). A procedure will be outlined below.

Inverting eqs. (4.6)

$$\eta = \chi\left(\frac{\xi_u - \xi_R - \psi\eta_R}{1 - \psi\chi}\right)$$

$$\xi = \left(\frac{\xi_R + \psi(\eta_R - \chi\xi_u)}{1 - \psi\chi}\right) \tag{4.7}$$

and furnishes the actual field coordinates.

The transformed description in $(\zeta, \bar{\chi}, \bar{\psi})$ space, where $\bar{\chi}(j)$ and $\bar{\psi}(i)$ are the normalized parameters

$$\bar{\chi} = \frac{\chi - \chi(1)}{\chi(j_m) - \chi(1)}$$

$$\bar{\psi} = \frac{\psi - \psi(1)}{\psi(i_m) - \psi(1)} \tag{4.8}$$

for each of the two grid directions, follows from eqs. (3.7) on application of

$$\frac{\partial}{\partial\zeta} = \frac{\partial}{\partial\zeta}$$

$$\frac{\partial}{\partial\eta} = \frac{\chi}{\eta(\chi(j_m) - \chi(1))}\frac{\partial}{\partial\bar{\chi}} + \frac{\psi^2}{(\xi - \xi_R)(\psi(i_m) - \psi(1))}\frac{\partial}{\partial\bar{\psi}} \tag{4.9}$$

$$\frac{\partial}{\partial\xi} = \frac{\chi^2}{\eta(\chi(j_m) - \chi(1))}\frac{\partial}{\partial\bar{\chi}} + \frac{\psi}{(\xi - \xi_R)(\psi(i_m) - \psi(1))}\frac{\partial}{\partial\bar{\psi}}$$

There results

$$E^*_{\zeta} + F^*_{\bar{\chi}} + G^*_{\bar{\psi}} + H^* = 0 \tag{4.10}$$

with the revised state vectors

$$E^* = \bar{E}$$

$$F^* = \frac{\chi(\bar{F} + \chi\bar{G})}{\eta(\chi(j_m) - \chi(1))}$$

$$G^* = \frac{\psi\,(\psi\bar{F} + \bar{G})}{(\xi - \xi_R)(\psi(i_m) - \psi(1))} \tag{4.11}$$

$$H^* = \bar{H} + \frac{\psi[\chi(j_m) - \chi(1)]\,F^* + \chi[\psi(i_m) - \psi(1)]G^*}{1 - \psi\chi} - \frac{\psi\bar{F}}{\xi - \xi_R} - \frac{\chi\bar{G}}{\eta}$$

An analgous grid system for the upper (expansion) side of the geometry is shown in Figure 7 and consists of

$$\chi = \frac{\eta - \eta_u}{\xi_u - \xi}$$

$$\psi_e = \xi - \eta\tan\beta_B \tag{4.12}$$

Or, on inverting

$$\eta = \frac{\chi(\xi_u - \psi_e)}{1 + \chi\tan\beta_B}$$

$$\xi = \frac{\chi\xi_u\,\tan\beta_B + \psi_e}{1 + \chi\tan\beta_B} \tag{4.13}$$

Again normalizing the grid level as in eqs. (4.8), the transformation to $(\zeta, \bar{\chi}, \bar{\psi}_e)$ space makes use of

$$\frac{\partial}{\partial \eta} = \frac{\chi}{\eta(\chi(j_m) - \chi(1))} \frac{\partial}{\partial \bar{\chi}} - \frac{\tan \beta_B}{\psi_e(i_m) - \psi_e(1)} \frac{\partial}{\partial \bar{\psi}_e}$$

$$\frac{\partial}{\partial \xi} = \frac{\chi^2}{\eta(\chi(j_m) - \chi(1))} \frac{\partial}{\partial \bar{\chi}} + \frac{1}{\psi_e(i_m) - \psi_e(1)} \frac{\partial}{\partial \bar{\psi}_e} \tag{4.14}$$

and results in

$$E^*_{e_\zeta} + F^*_{e_{\bar{\chi}}} + G^*_{e_{\bar{\psi}e}} + \bar{H}_e = 0 \tag{4.15}$$

with

$$E^*_e = \bar{E}$$

$$F^*_e = \frac{\chi\,(\bar{F} + \chi \bar{G})}{\eta(\chi(j_m) - \chi(1))}$$

$$G^*_e = \frac{\bar{G} - \tan\beta_B \bar{F}}{\psi_e(i_m) - \psi_e(1)} \tag{4.16}$$

$$H^*_e = \bar{H} - \left[\frac{(1 + 2\chi \tan\beta_B)\bar{G} + \tan\beta_B \bar{F}}{\xi_u - \psi_e} \right]$$

Grid Parameters

The grid systems permit adjustment of the i and/or j locii so that some mesh line(s) approximately parallel any expected shock discontinuities for $M_\infty \leq M_D$. Near on-design a shock lying along $\xi \approx \xi_T$ corresponds to assuming $\xi_R \approx \xi_T$, for example. Off-design attached shocks correspond to $\eta_R \geq \eta_T$ with ψ = constant matching either the oblique shock trace near the edge, or linearly approximating the entire shock trace over the entire $0 \leq \eta \leq 1$ interval.

For detached shocksfrom the side edge, choices for η_R and η_N can be made such that the j_N grid line approximates the normal to the surfaces, and simultaneously the $\psi = 0$ mesh line approximates the central ($\eta << 1$) shock location. From eq. (4.6)

$$X_M = \frac{\eta_M}{\xi_u - \xi_M} = \frac{1}{\left(\frac{\xi_u}{\eta_M} - \tan \beta_B\right)} = (j_M - 1.5)\,\Delta\chi$$

$$X_N = \frac{\eta_N}{\xi_u - \xi_N} = \frac{1}{\left(\frac{\xi_u}{\eta_N} - \tan \beta_B\right)} = (j_n - 1.5)\,\Delta\chi \qquad (4.17)$$

Therefore for a given number of grid lines outboard of the edge, or equivalently, if given

$$j_{mn} = \frac{j_m - 1.5}{j_n - 1.5} \qquad (4.18)$$

the location ξ_u is

$$\xi_u = \xi_R \left(\frac{j_{mn} - 1}{j_{mn} - \frac{\eta_M}{\eta_N}} \right) \tag{4.19}$$

Conversely, a given (ξ_u/ξ_R) implies some j_{mn}, which necessarily should be restricted to levels leading to integer values of

$$j_m = 1.5 + (j_n - 1.5)\, j_{mn} \tag{4.20}$$

which poses no difficulty. Essentially, eq. (4.19) shows that

$$j_{mn} \geq \frac{\eta_M}{\eta_N} \geq 1 \quad \text{implies } \infty \geq \xi_u \geq \xi_R \; (= \xi_M) \tag{4.21}$$

and the ξ_u such that j_N is normal to the upper (expansion) surface extended (i.e. $\overline{TM}$) fixes $\chi(j_N) = \tan\beta_B$, and from eq. (4.17) is then

$$(\xi_u)_N = \frac{\eta_N}{\sin\beta_B \cos\beta_B} \tag{4.22}$$

In that case also

$$\chi(j_M) = \frac{\sin\beta_B \cos\beta_B}{\frac{\eta_N}{\eta_M} - \sin^2\beta_B} \tag{4.23}$$

The compression side grid extends to the intersection of the $(i,j) = (1,j_m)$ meshlines or to (point P, Figure 6)

$$\eta_{max} = \frac{(\xi_u - \frac{\tan\delta}{s^*})\chi_M}{1 + g'\chi_M} \tag{4.24}$$

The j line procedure may be summarized as follows:

a) Assume η_N in order to fix the mesh point location outboard of the airfoil edge, and from eq. (4.22) the approximate $(\xi_u)_N$ for a normal meshline follows.

b) Assume ξ_R/ξ_T to fix η_R from eq. (3.30), and also $\eta_M (= \xi_M/\tan\beta_B)$:

c) Assume j_n and fix integer value of j_m from eqs. (4.19), (4.20).

The i line procedure is straightforward from eqs. (4.6) and (4.12).

$$\psi(i) = \left(\frac{\xi_{\eta=0} - \frac{\tan\delta}{s^*}}{\eta_R}\right) + \psi(1) = (i-1)\Delta\psi + \psi(1)$$

$$\psi_e(i) = \xi_{\eta=0} = (i-1)\Delta\psi_e + \psi_e(1) \tag{4.25}$$

in which $\psi(1) = -g'$ and $\psi_e(1) = 0$. The choice of a maximum $|\xi_{\eta=0}|$ based on expected shock strengths (see, e.g., eq. (2.7)) then fixes both $\psi(im)$, $\psi_e(im)$ levels.

Boundary Conditions

Boundary conditions are of four kinds: surface, symmetry, overlapping computational grids and grid external field edges. The latter were maintained at free stream property levels (along the i_m and j_m boundaries), excepting for certain attached shock cases that took advantage of the known uniform regions of compression near the edges to reduce consideration to solely the lower region.

The surface boundary condition from eq. (3.31) is

$$\left(\frac{v}{u}\right)_w = \frac{\left(\frac{w}{u}\right)_w - \tan\delta}{g'} \tag{4.26}$$

for the compression surface with $g'_\ell = (\tan\beta_D - \tan\delta) / s^*$. The same form applies for the expansion side with δ taken equal to zero; i.e.

$$\left(\frac{w}{v}\right)_w = \tan\beta_B \tag{4.27}$$

In practice such boundary conditions were enforced after each integration step as the calculation proceeds and eq. (4.26) indicates some freedom in choosing u, v, or w as the basis for such adjustments. Since $q^2 = u^2+v^2+w^2$ (and omitting $(\ )_w$ for convenience), eq. (4.26) implies quadratic equations for v and w:

$$\begin{aligned} aw^2 - 2bw + c + (q^2 - u^2)(1 - g'^2) &= 0 \\ av^2 + 2bg'v + c &= 0 \end{aligned} \tag{4.28}$$

Here

$$\begin{aligned} a &= 1 + g'^2 \\ b &= u\tan\delta \\ c &= u^2(1 + \tan^2\delta) - q^2 \end{aligned} \tag{4.29}$$

and therefore

$$\begin{Bmatrix} w \\ v \end{Bmatrix} = \frac{\{\begin{smallmatrix} 1 \\ -g' \end{smallmatrix}\} u\tan\delta \pm \{\begin{smallmatrix} g' \\ 1 \end{smallmatrix}\}\sqrt{(v^2+w^2)(1+g'^2)-(u\tan\delta)^2}}{1+g'^2} \tag{4.30}$$

The contents of the radical dictate that

$$q^2 - u^2 = v^2 + w^2 = k_1 \frac{(u\tan\delta)^2}{1+g'^2} \qquad [k_1 \geq 1] \tag{4.31}$$

where the $k_1 = 1$ limit corresponds to a minimum $(q^2 - u^2)$ level for which the specific (v,w) pair from eq. (4.30) are

$$\begin{cases} w_{min} = \dfrac{u\tan\delta}{1+g'^2} > 0 \\ \\ v_{min} = -g'w_{min} < 0 \end{cases} \tag{4.32}$$

Thus

$$\begin{Bmatrix} w \\ v \end{Bmatrix} = w_{min}\left[\begin{Bmatrix} 1 \\ -g' \end{Bmatrix} \pm \begin{Bmatrix} g' \\ 1 \end{Bmatrix}\sqrt{k_1 - 1}\right] \tag{4.33}$$

and the radical sign choices for a given u are clarified in Figure 8; i.e. a positive sign results in $w > w_{min}$ and $v > v_{min}$, but $v > 0$ only if $w > u\tan\delta$. Eqs. (4.26) and (4.32a) show that the locus of min points as u varies is given by

$$g'w_{min} = -v_{min}$$

as is also clear from the slopes of the constant u line and its normal.

Point C in Figure 8, i.e. (v,w) = (0, utanδ), is the reentrant corner condition as well as the entire compression surface condition if on-design. For off-design eq. (4.26) implies an interval $0 \leq |v| \leq |v_T|$ along the ACB line for some constant u. An initial guess for the field does not necessarily correspond to the correct interval, although in the attached shock case the exact interval limits are predetermined. However, in either case the algorithm eqs. (4.2), (4.3) does not ensure consistent state properties for eq. (4.26) as the calculation proceeds.

Thus the E* vector as given by a corrector algorithm, eq. (4.3), may result in a (v,w) pair that does not lie on the u = constant locus (Figure 8), and in fact may lie within the interior of the $k_1 = 1$ circle. That the possibility is of importance may be realized by noting that g' can be small, so that the points min and C are in fact close together. Continuation of the calculation requires distinct procedures for the $k_1 \gtrless 1$ situations. If $k_1 > 1$ the imposed boundary condition after each corrector step is given by eq. (4.33); if $k_1 < 1$ the effective k_1 may be ammended to that satisfying eq. (4.33) for v, namely

$$k_1 = 1 + \left(\frac{v - v_{min}}{w_{min}} \right)^2 \tag{4.35}$$

and w then follows from eq. (4.33). This procedure effectively increases w without constraint on the sign of v.

Alternatively, with v as the basis after each corrector step

$$u = \frac{-2vg'\tan\delta + \sqrt{(2vg'\tan\delta)^2 - (1+\tan^2\delta)[(1+g'^2)v^2-q^2]}}{1+\tan^2\delta}$$

$$w = u\tan\delta + vg' \qquad (4.36)$$

for the compression surface, and

$$u = \sqrt{q^2 - \left(\frac{v}{\cos\beta_\beta}\right)^2}$$

$$w = v\tan\beta_\beta \qquad (4.37)$$

for the expansion side. Both eqs. (4.33) and (4.36), (4.37) were used successfully, the former for the attached shock and the latter for the detached shock applications.

The symmetry plane ($\eta = 0$) condition imposes a reflection of the $j = 1$ mesh line properties properly interpolated to take acccunt of the diverging grid in the (η,ξ) plane. With ξ and ψ from eqs. (4.7) and (4.25), and $\chi(j) = \bar{+}\,\Delta\chi/2$ for $j = 1.2$ the interpolation follows from (Figure 9) the weighting

$$\frac{\delta\xi}{\Delta\xi} = \frac{\xi(i,2) - \xi(i,1)}{\xi(i+1,2) - \xi(i,2)} \qquad \gtrless 0 \text{ if } \psi(i) \lesseqgtr 0 \qquad (4.38)$$

E.g.,

$$p(i,1) = p(i,2) - [p(i+1,2) - p(i,2)]\frac{\delta\xi}{\Delta\xi} \qquad (4.39)$$

Similarly, ρ, u, w symmetry and v asymmetry may be imposed at all (i,1) points. For the expansion side ($\xi < 0$)

$$\Delta\xi = \xi(i,2) - \xi(i+1,2) \tag{4.40}$$

and

$$p(i,1) = p(i,2) - [p(i,2) - p(i+1,2)] \frac{\delta\xi}{\Delta\xi} \tag{4.41}$$

Outboard of the tip ($\eta > 1$) either symmetry or overlapping grid interpolations were imposed. For attached shock conditions TM (Figure 6) may be considered to be a symmetry plane and the grid values for p, ρ, u, and $(v^2 + w^2)$ were upgraded after each sweep of the entire mesh region so as to impose that symmetry. For detached shock conditions the overlapping grid results were used to upgrade the boundary conditions along either TM or TP (for the upper or lower region respectively) based upon the results of a set number of sweeps for the prior integration region.

In the symmetry case, for each $j_n \leq j \leq j_m - 1$ the ψ corresponding to each $\chi(j)$ along TM is

$$\psi = \frac{(\xi_u - \xi_R)\ \chi \tan\beta_B - \xi_R}{\eta_R + \chi(\eta_R \tan\beta_B - \xi_u)} \tag{4.42}$$

which fixes the maximum (integer) i for a given j in region TPM on use of eq. (4.25). Since the normal to TM has the slope $(-\cot\beta_B)$ the symmetrically opposite point corresponds to

$$\chi_{opp} = \frac{\eta_{opp}}{\xi_u - \xi_{opp}}$$

$$\psi_{opp} = \frac{\xi_{opp} - \xi_R}{\eta_R - \eta_{opp}} \tag{4.43}$$

where

$$\eta_{opp} = \frac{\xi_i \tan\beta_B + \eta_i}{1 + \tan\beta_B \tan(2\beta_B - \beta_i)}$$

$$\xi_{opp} = \eta_{opp} \tan(2\beta_B - \beta_i) \tag{4.44}$$

and $\beta_i = \tan^{-1}(\xi_i/\eta_i)$. Thus

$$p(i,j) = p(i+1,j) + [p(i+1,j)-p(i+1,j-1)] \frac{\chi_{opp}-\chi(j)}{\Delta\chi} + [p(i+1,j)-p(i+2,j)] \frac{\psi(i+1) - \psi_{opp}}{\Delta\psi} \tag{4.45}$$

with similar relations for $p(i,j)$, $u(i,j)$, and v_{opp}, w_{opp}. Then

$$v(i,j) = v_{opp} \cos 2\beta_B + w_{opp} \sin 2\beta_B$$

$$w(i,j) = v_{opp} \sin 2\beta_B - w_{opp} \cos 2\beta_B \tag{4.46}$$

such that $v^2 + w^2 = v_{opp}^2 + w_{opp}^2$. The levels from eqs. (4.45),(4.46) are then associated with all i equal or less than that from (4.42) to complete the grid specification throughout TPM.

The procedure for the overlapping grid case is similar to the above. Partially converged results for the lower region, e.g., provide conditions along TM as the tip condition for the upper region integration. Eq. (4.42) defines both ξ_{TM} for a given $j_n \leq j \leq j_m-1$, and with eq. (4.25) the adjacent i values (i^+, i^-) bracketing the point on the TM locus. Then

$$p_{TM} = p_{i-} + (p_{i+} - p_{i-})\left(\frac{\xi_{TM} - \xi_{i-}}{\xi_{i+} - \xi_{i-}}\right) \tag{4.47}$$

and similarly for the other state properties. On completion of partial convergence for the upper region, the same procedure establishes values along TP as the subsequent lower region condition. In that case the ψ_e corresponding to each $\chi(j)$ point on TP is

$$\psi_e = \frac{\tan\delta(1 + \chi[\tan\beta_B - \xi_u])}{s^*(1 + g'\chi)} \tag{4.48}$$

Decoding

The flow variables must be decoded from E^* or E_e^* results in order to reconstruct F^*, G^*, H^* after each predictior and corrector step. From eqs. (3.2) and (3.4)

$$u = \frac{E_2 + \sqrt{E_2^2 - 4k(1-k)(E_1^2 - E_3^2 - E_4^2)}}{2(1-k)E_1}$$

$$v = \frac{E_3}{E_1} \tag{4.49}$$

$$w = \frac{E_4}{E_1}$$

$$\rho = \frac{E_1}{u}$$

$$p = \frac{E_1}{u}\left(1-u^2 - \left(\frac{E_3}{E_1}\right)^2 - \left(\frac{E_4}{E_1}\right)^2\right) \qquad (4.49)\ \text{(continued)}$$

The sign choice for the radical follows from consideration of small v and w, or that u remains supersonic. Once again, however, the finite difference procedure does not ensure consistent E_i values after each $\Delta\zeta$ step. If R is the contents of the radical in eq. (4.49) there are two implied physical constraints:

$$R \geq 0$$

$$\frac{E_2 + \sqrt{R}}{2(1-k)E_1} \leq 1 \qquad (4.50)$$

The first ensures real u and the second that $u \leq q \leq 1.0$.

I.e.,

$$\frac{E_2}{E_1} \begin{cases} \geq \left(\frac{E_i}{E_1}\right)_{min} = (4k(1-k)[1 - (q^2-u^2)])^{1/2} \\ \leq \left(\frac{E_2}{E_1}\right)_{max} = 1 - k(q^2 - u^2) \end{cases} \qquad (4.51)$$

is the permissible E_2/E_1 range for the calculation to continue.

These bounds are of special importance for undershoot/overshoot behavior near shock locations, for near sonic speed, and at high Mach numbers. In the present computations both the $R < 0$ and $p < 0$ possibilities were monitored and corrected for by modification of E_2 to constrain the system to physically meaningful results during the asymptotic approach. Such modifications proved necessary when starting with arbitrary initial conditions but the need does decay (i.e. in terms of number of necessary modifications) as the integration proceeds.

Figure 10 illustrates the limits of eqs. (4.51) and also includes the E_2/E_1 variation with M for the special case of $v = w = 0$; i.e. in the latter case

$$\left(\frac{E_2}{E_1}\right) = \frac{\frac{\gamma-1}{2\gamma} + \frac{\gamma-1}{2} M^2}{\left[\frac{\gamma-1}{2} M^2 \left(1+ \frac{\gamma-1}{2} M^2\right)\right]^{1/2}} \qquad (v = w = 0) \tag{4.52}$$

and varies between $(\gamma^2-1)^{1/2}/\gamma = 0.6999$ and unity for $1 \leq M \leq \infty$. In general the minimum condition corresponds to

$$u^2 \geq \frac{k(1-q^2)}{1-2k} \tag{4.53}$$

and the $v = w = 0$ special case implies $u = \sqrt{k/(1-k)}$ or sonic speed, consistent with eq. (4.52).

Solution Procedures

The integration of the field proceeds from assumed initial conditions at the arbitrary axial plane $\zeta = 1$. Free stream conditions suffice but an

appreciable reduction in computation time results with either imposed shock layer flow variables which are based on psuedo on-design conditions or the continuation from a previously determined solution as the conditions closely approximating those next of interest. An attached shock evaluation would therefore typically follow by taking a ΔM_∞ increment from an already converged solution for some M_∞, starting originally with the on-design case. At each ζ the predictor and corrector computations from eqs. (4,2), (4.3) imply decoded values applicable at $\zeta + \Delta\zeta$, boundary conditions are then enforced, and the steps are repeated until uniform conical conditions are achieved at some downstream ζ.

For known attached shock conditions (η_R, ξ_R) was placed at the side edge and ξ_u moved to ∞ such that the j mesh lines formed a set parallel to the $\eta = 0$ centerline. In such cases only the lower (compression) region need be considered and the outermost grid boundary (j_m) was displaced inboard so as to lie within the uniform flow region between the tip and the post shock Mach cone from the apex (Figure 11). This arrangement provides an i grid line which parallels the shock in the uniform region and places the singular $(\ )_R$ origin external to the computational domain. However, free stream conditions are no longer appropriate along j_m and were replaced in such cases with the simple floating uniformity condition $p(i,j_m) = p(i,j_m-1)$, etc. The tip region properties therefore were self adjusting to a final sweptback edge compression level.

For detached shock conditions, as well as when check computations of the attached shock case were considered the alternating grid procedure was employed with partial convergence of the respective upper and lower regions carried out successively. Initial conditions consisted of finite

shock layer approximations extended over the entire lower (compression) region and the tip portion of the upper region. This is of some importance for small δ since the overlap region TMP in that case may itself be small.

5. NUMERICAL RESULTS

Computations were carried out for two basic caret sections with $\beta_D = 35°$ and $\delta = 5.746°$ and $22.181°$ (Table 3). The geometries correspond to on-design flows at $M_D = 2$ and 4 respectively and in each case both $s^* = 1$ and 2 were considered. For the $M_D = 2$ geometry both the attached and detached (alternating) algorithms were used for some of the attached side edge cases as a consistency check on the procedures.

A measure of the shock capturing behavior is shown in Figures 12 a,b for several $M_\infty < M_D$ at each of the M_D levels. The pressure distribution through the shock layer adjacent to the symmetry plane ($\eta = 0.01$) for attached side edge cases exhibits fairly uniform conditions both within and external to the shock layer. Overshoot and undershoot was always present upstream and downstream of the shock location with a numerical adjustment to the pressure jump occurring over 3 to 4 mesh widths. The major shock jump was over two mesh widths as indicated by the dashed lines.

Lower (compression) surface pressure coefficient distributions appear in Figures 13 a, b, c. Figures 13 a and c contain only attached side edge cases; Figure 13 b includes those for side edge detachment. The attached algorithm imposed the uniformity condition at $\eta > \eta_s$ (Figure 11) and led to converged solutions in good agreement with the theoretical pressure levels (indicated by horizontal lines in Figures 13) for the uniform edge regions, and with break points in the distribution quite consistent with the expected η_s locations. The detached shock cases ($M_\infty = 1.3, 1.4$ in Figure 13 b) achieved near edge pressures comparable with post shock levels for a symmetric shock lying at $1 \lesssim \eta < 1.06$.

E.g., for the $M_\infty = 1.3$ case $0.38 \lesssim C_p \lesssim 0.41$ is implied behind such shocks when lying outboard of the side edge.

Upper (expansion) surface pressures for detached cases indicated free stream levels over the entire surface span. The expansion about the side edge rapidly weakens the nearby shock and resulted in a pressure field for $\eta < 1$ within approximately one percent of the free stream pressure. Nevertheless, the upper region is of some importance as a boundary in the detached region which allows for the rapid changes.

Shock jump locii in the crossplane are shown in Figures 14 a, b, c, the geometry being inverted. All points were obtained as midpoint estimates of the greatest pressure jump location along each grid line (e.g., as in Figure 12). Solid lines in Figures 14 a, c are the theoretical attached shock traces for $\eta > \eta_s$ (Figure 11). The interpolated points are in good agreement with such directions and maintain the same direction for an additional interval $\eta < \eta_s$ in which the edge conditions persist away from the surface and exterior to the η_s Mach cone.

The dashed line in Figures 14 a, c indicates the limiting shock directions for post shock supersonic flow normal to the edge. For these geometries (Table 3) the condition corresponds to $M_\infty = 1.428$ and 2.326 for the (δ, β_D) pair examples. The $M_\infty = 1.4$ case in Figure 14 b represents a condition very close to the classical detachment level ($M_\infty = 1.41$); $M_\infty = 1.3$ is a fully detached case from the side edge and is approaching close to the apex detachment limit of $M_\infty = 1.26$.

In both Figures 14 a and c the point scatter increases appreciably as the shock traces depart from the computational grid directions,

i.e. for $\eta \to 0$ regions and for increasing $|M_D - M_\infty|$, with an indicated waviness on the order of the grid scale. For the detached $M_\infty = 1.3$ case the grid scale was approximately halved to confirm the effect for a widely bulging shock.

The $M_\infty = 1.4$ case closely adhered to the limiting shock direction for a supersonic side edge with virtually no discernible pressure jump in the field on the expansion side. For $M_\infty = 1.3$ the detachment is quite clear and an appreciably weakened shock is visible on the expansion side. Two Mach cone traces are shown for the upper (expansion) surface in Figure 14 b. One is the apex Mach cone for $M_\infty = 1.3$ which extends to about $\eta = 0.6$. The other is the free stream Mach cone disturbance envelope limit from the side edge which is tangent to the Mach cone centered on the apex. The latter limit makes clear the rapid weakening of the shock in the close vicinity of the edge and the reasonably parallel shock trace extending into the upper field. This appears to be consistent with the virtually undisturbed surface pressures found for that surface as mentioned above. The detachment scale itself remains small ($\eta \simeq 1.05$) despite the close approach to complete detachment at $M_\infty = 1.26$.

Figures 15, 16, and 17 summarize the overall force coefficients for lift and drag based on

$$C_L = \frac{2}{\gamma M_\infty^2} \left(\int_0^1 \left(\frac{p}{p_\infty}\right)_\ell d\eta - \int_0^1 \left(\frac{p}{p_\infty}\right)_e d\eta \right)$$

$$C_D = \frac{2\tan\delta}{\gamma M_\infty^2} \left(\int_0^1 \left(\frac{p}{p_\infty}\right)_\ell d\eta - \frac{p_{base}}{p_\infty} \right) \qquad (5.1)$$

with (p_{base}/p_∞) taken to be zero. The base pressure level can make a substantial contribution to a drag reduction, of course, depending on the Mach number level. For example, on-design the extremes of (p_b/p_∞) for $M_D = (2,4)$ imply the range $C_D = (0.0492, 0.2157)$ to $(0.0133, .1793)$. For $M_\infty = 2$, C_L/C_D increases from 2.68 to 9.94, and for $M_\infty = 4$ from 2.04 to 2.45 for (p_b/p_∞) in the range 0. to 1.

Comparison is made between results from the attached and detached algorithms in Figures 15 and 16 for $s^* = 2$ and $M_\infty = 1.7$ and 1.5. The C_L disagreement for the latter may be attributed to edge effects on the upper surface under near detachment conditions, as evident in the C_D unaffected agreement for the same cases. For relatively smaller η_s $(M_\infty = 1.7)$ the alternating procedure proved to be very consistent.

The coefficient variation in the detachment range of M_∞ shows no abrupt lift losses due to edge effects. One computation was purposely carried out beyond the apex detachment limit as a consistency check; for $M_\infty = 1.2$ a sharp decrease in C_L was indicated (approximately 20%) on starting from the $M_\infty = 1.3$ solution and completing several hundred $\Delta\zeta$ steps. The lift to drag ratio decreases smoothly from 2.68 to 2.48 for attached shocks at $2.0 \geq M_\infty \geq 1.50$ and averages $2.47 \pm 1\%$ in the detached M_∞ interval.

The detachment evaluations depend critically on an appropriate initial condition in the overlapping mesh regions PTM, Figure 6. For thin geometries, such as the $\delta = 5.75°$, $M_D = 2$ example cited here, it is essential that initial conditions along the boundaries PT and TM be representative of a shock layer, and that the overlap be sufficient to

allow description of the gradients in that region. For $P \to M$, i.e. an extremely thin geometry, the alternating algorithm would effectively require a match of derivatives along the common interface. In this sense the calculations for $\delta \simeq 5°$ are a good test of the capabilities of overlapping mesh regions chosen so as to conform to specific surfaces and shock layer segments. It is to be expected that "thick" caret solutions, e.g., would converge somewhat faster and be less sensitive to initial field choices. Virtually all of the present results were obtained, however, by stepping M_∞ to a new level and proceeding from the previous field.

6. CONCLUSION

The study has applied shock capturing finite difference procedures to waverider configurations of the caret type. It is fairly evident that a grid definition for crossplane sections with cornersand concave regions poses special constraints. For off-design Mach numbers such that shocks remain attached at side edges it has been shown that computational grids may be chosen to conform to crossplane surface and symmetry planes while simultaneously providing approximate but adequate matching with anticipated shock traces. For the Mach number interval corresponding to edge detached to apex detached flow fields the interacting compression and expansion portions of the field may be evaluated by similar specialized grid regions of relatively small overlap.

The suggested alternating algorithm requires sequencing the separate integrations for each such region, with interpolative adjustment of the boundary conditions in the edge region as the asymptotic integration proceeds. The evaluations demonstrate the utility of such an alternating algorithm for relatively small regions of overlap that are implied by thin configurations.

While no attempt was made to refine grid scales so as to define the flow details in the detached near edge region, the overall force coefficient results make clear that sharp discontinuities in behavior are not present in the transition from attached to detached side edge conditions.

APPENDIX 1

COMPUTER CODE

A code glossary, flow chart and listing are given below for the detached shock field algorithm. Computations were carried out on the M.I.T. Multiplexed Information Computing Service (MULTICS) system which makes use of a Honeywell 6180 computer in a time sharing mode. Computer time per grid point was 2.34×10^{-3} seconds and solution time varied from 10 minutes for nearly on-design to two hours for a fully detached case.

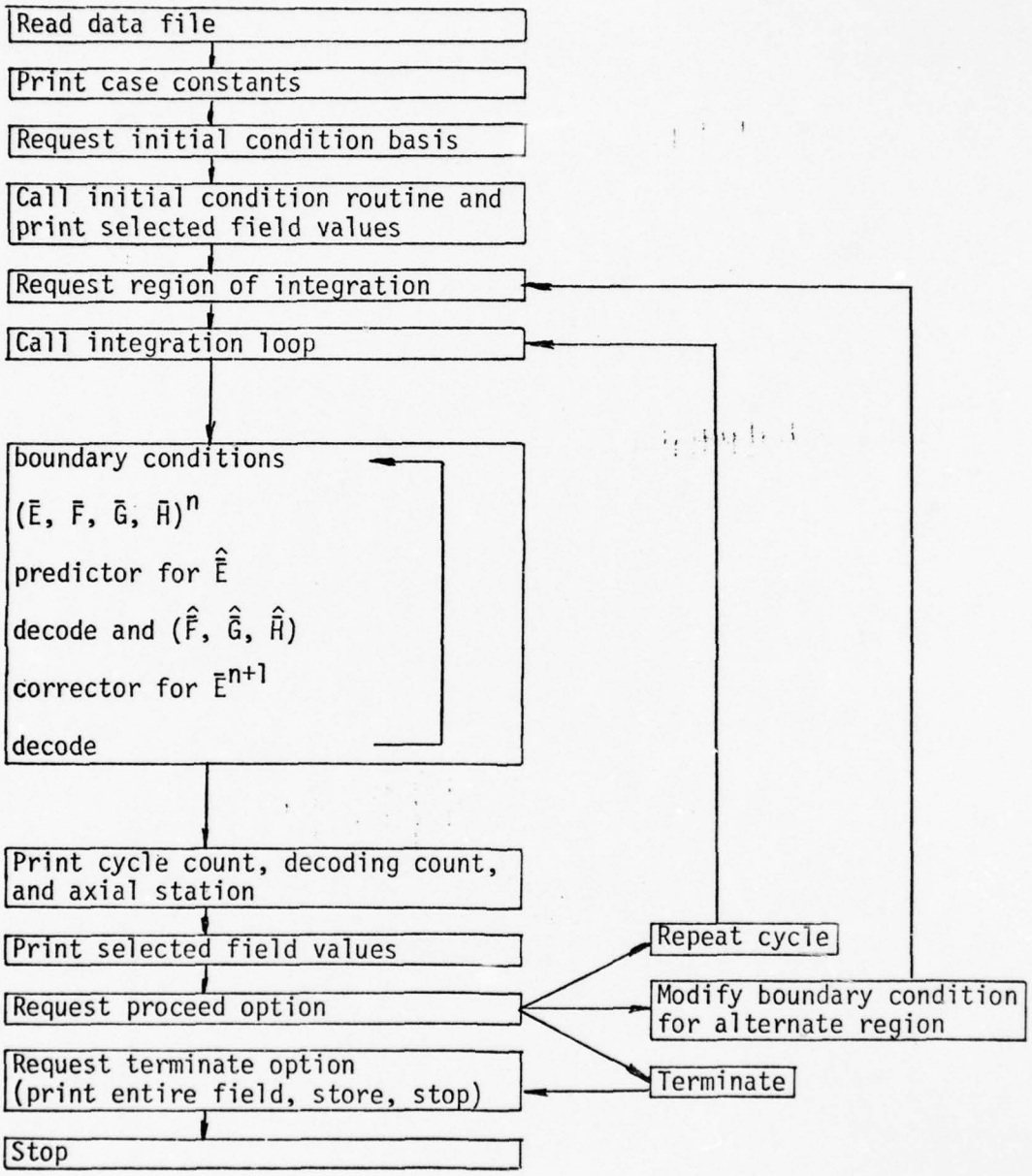
Specific grid constants are summarized in Table 4 for the cases considered. Stable solutions were completed with bet = 0.05. Numerical instabilities were evident for bet > 0.10.

Code Glossary

a	$(1 + \tan^2\delta)$
ajmn	$(j_m-1.5)/(j_n-1.5)$
ak	$(\gamma-1)/2\gamma$
akl	η_M
ak4	initial condition shock layer weighting factor
al	$\Delta\zeta/\Delta\bar{\chi}$
am	M_∞
amd	M_∞ on design
b	vh tanδ
be	β_D
bet	$\Delta\zeta/\Delta\bar{\psi}$
chi	χ
d	$(1+h^2)v^2-q^2$
dz	$d\zeta$
dcc	$\Delta\chi$
dcp	$\Delta\psi$
del	δ
del chib	$\chi(j_m) - \chi(1)$
del psib	$\psi(i_m) - \psi(1)$
dxir	$\delta\xi/\Delta\xi$
e	$\bar{E}$
et	η
etb	η along β_B locus
etl	η_L
etr	η_R
f	$\bar{F}$
g	$\bar{G}$
ga	γ
h	$\bar{H}$
hh	$(\tan\beta_D - \tan\delta)/s^*$
im, iml	i_m
iii	i_m-1
j_m, jml	j_m

jn	j_n
jnl	$j_n + 1$
jjj	$j_m - 1$
ni	cycle count
p	p
pi	π
pp	$(p/p_o)_\infty$
psi	ψ
pl	p/p_{o_∞}
p2	p/p_∞
q	q
qq	q^2
r	ρ
rr	$(\rho/\rho_o)_\infty$
rl	ρ/ρ_{o_∞}
s	s*
tbb	$\tan\beta_B$
tbe	$\tan\beta_D$
tbbi	$\tan(2\beta_B - \beta_i)$
u	u
uu	$(q/q_{max})_\infty$
v	v
w	w
xi	ξ
xib	ξ along β_B locus
xir	ξ_R, ξ_M
xiu	ξ_u
xid	ξ_{im} for $\eta = 0$
z	ζ

FLOW CHART


Read data file
Print case constants
Request initial condition basis
Call initial condition routine and print selected field values
Request region of integration
Call integration loop
boundary conditions
$(\bar{E}, \bar{F}, \bar{G}, \bar{H})^n$
predictor for $\hat{\bar{E}}$
decode and $(\hat{\bar{F}}, \hat{\bar{G}}, \hat{\bar{H}})$
corrector for $\bar{E}^{n+1}$
decode
Print cycle count, decoding count, and axial station
Print selected field values
Request proceed option
Repeat cycle
Modify boundary condition for alternate region
Terminate
Request terminate option (print entire field, store, stop)
Stop

```
c                 detached main
c
      dimension p(42,42),r(42,42),u(42,42),v(42,42),w(42,42)
      dimension et(42,42),xi(42,42),chi(42),psi(42)
      dimension pl(42,42),rl(42,42),ul(42,42),vl(42,42),wl(42,42)
      dimension etl(42,42),xil(42,42),chil(42),psil(42)
      dimension e(42,42,4),f(42,42,4),g(42,42,4),h(42,42,4)
      dimension e1(42,42,4)
      common /c1/ p,r,u,v,w
      common /c2/ ga,am,be,pi,amd
      common /c3/ hh,dcc,dcp,tde,dccl,dcpl
      common /c4/ im,jm,iml,jml
      common /c5/ ak9,ak4
      common /c6/ jjn,a,xiu,tbb,etr
      common /c7/ iii,jjj,iiil,jjjl,jn1,jt,jt1
      common /c8/ xi, et,xil,etl
      common /c9/ jn
      common /c10/ xir,chi,psi,chil,psil
      common /c11/ ak
      common /c13/ s
      common /c14/ e1,e,f,g,h
      common /c15/ dz,al,bet,z
      common /c16/ delchib,delpsib,delchil
      common /c17/ negr,negp
      common /c18/ pl,rl,ul,vl,wl
      common /c19/ nit
      common /c20/ ja,jb
      namelist /nam/ ga,am,s,be,del,bet,jn,etal,
     &                ak1,ak2,ni,amd,ak3,nit,ak4
      rewind 22
c                 constants, coordinates
      read (22,nam)
      ak   = (ga-1.)/(2.*ga)
      pi   = 4.*atan(1.)
      tde  = tan(del/180.*pi)
      a    = 1.+tde**2
      tbe  = tan(be/180.*pi)
      tbb  = tbe/s
      hh    = (tbe-tde)/s
      xir  = ak1*tbb
      etr  = (xir - (tde/s))/hh
      xiu  = 2.*etal/(sin(2.*atan(tbb)))
      ajmn = ((xiu/etal)-tbb)/((xiu/ak1)-tbb)
      ak9  = 1.5 + (jn-1.5)*ajmn
      jm   = ifix(ak9)
      im   = jm
      jml  = jm
      iml  = im
      ajmn = (jm-1.5)/(jn-1.5)
      xiu  = xir*(ajmn-1.)/(ajmn-(ak1/etal))
      xicl = ak2*xir
      dcc  = 1./((xiu/ak1-tbb)*(jm-1.5))
      dcp  = (xicl-(tde/s))/etr/(im-1.)
      dccl = dcc
      dcpl = ak3/(iml-1.)
      jjjl = jml - 1
      iiil = iml - 1
      jn1  = jn + 1
      jjj  = jm-1
      iii  = im-1
```

```
      jjn  = jr-1
      dz   = bet/iii
      al   = dz*jjj
      print 100,  am,del,be,(180./pi)*atan(tbb),s,
     &            etal,ak1,etr,xiu,xicl,
     &            ak2,dcc,dcp,ak4,
     &            ak3,dccl,dcpl,
     &            jn,jm,im,
     &            nit,jml,iml,
     &            al,bet,amd
c
      do 11  i = 1,im
      psi(i)   = (i-1. )*dcp - hh
      do 10  j = 1,jm
      chi(j)   = (j-1.5)*dcc
      ak9      = 1. - chi(j)*psi(i)
      et(i,j)  = chi(j)*(xiu-xir-etr*psi(i))/ak9
10    xi(i,j)  = (xir + (etr-xiu*chi(j))*psi(i))/ak9
11    continue
      do 204 j = 10,jjj
      ak9  = et(1,j) - 1.
      jt1  = j
      if (ak9.gt.0.) go to 205
204   continue
205   jt   = jt1 - 1
      print 146, jt
      print 120
      delchib  = chi(jm) - chi(1)
      delpsib  = psi(im) - psi(1)
      do 61  ii = 1,iml
      psil(ii)  = (ii-1.)*dcpl
      do 62  jj = 1,jml
      chil(jj)  = (jj-1.5)*dccl
      ak9       = 1. + chil(jj)*tbb
      etl(ii,jj)= chil(jj)*(xiu-psil(ii))/ak9
62    xil(ii,jj)= (chil(jj)*xiu*tbb + psil(ii))/ak9
61    continue
      delchil   = chil(jml) - chil(1)
      call prcoord
c                 initial conditions: -0+  disc,des,m
      print 102
      read 108,l
      call ic(l)
c                 integration
      z    = 0.
      kc   = 0
208   print 147
      read 108,l
23    do 21 k2 = 1,ni
      kc   = kc + 1
      ja   = jt
      jb   = jt - 1
      do 207 k1 = 1,nit
      call integrate(l)
207   continue
      ja   = 2
      jb   = 1
      call integrate(l)
21    continue
      print 120
```

```
      print 105, kc,nit
      print 124, negr,negp
      negr  = 0
      negp  = 0
      z     = z + dz*ni
c               not converged: -0+ mancyc,stop,selmod; -0+ prfin,stop,write
      rewind 9
      write (9) p,r,u,v,w,pl,rl,ul,vl,wl
24    print 106,z
      call pr
      print 107
      read 108,ll
      if (ll)  23,25,26
26    call mod(1)
      go to 208
25    print 109
      read 108,ll
      if (ll) 49,50,51
49    print 135
      call prfin
51    continue
50    stop
c
100   format ("m   =",f6.2," delta=",f8.4," betad=",f8.4," betab=",
     &          f8.4," b*   =",f5.1,/,"etal=",f6.2," etam =",f8.4,
     &          " etar =",f8.4," xiu  =",f8.4," xicl =",f8.4,/,
     &          "ak2 =",f6.2," delchi=",f8.4," delpsi=",f8.4," ak4  =",f8.4,/,
     &          "ak3 =",f6.2," delchl=",f8.4," delpsl=",f8.4,/,
     &          "jn  =",i4,4x,"jm   =",i4,6x," im =",i4,/,
     &          "nit =",i4,4x,"jml  =",i4,6x," iml =",i4,/,
     &          "al  =",f6.2," bet  =",f8.4," amd  =",f8.4,/)
102   format ("fix i.c.")
105   format ("iters=",i7,i7)
106   format ("ln x =",f8.2)
107   format ("proceed option")
108   format (i2)
109   format ("terminate option")
120   format (2x)
135   format (/,/,/)
124   format ("negr =",i6,2x,"negp =",i6)
146   format ("jt =",i7)
147   format ("fix region")
c
      end
c
c
c
c
      subroutine ic(1)
c               initial conditions
      dimension p(42,42),r(42,42),u(42,42),v(42,42),w(42,42)
      dimension pl(42,42),rl(42,42),ul(42,42),vl(42,42),wl(42,42)
      common /c1/ p,r,u,v,w
      common /c2/ ga,am,be,pi,amd
      common /c3/ hh,dcc,dcp,tde,dccl,dcpl
      common /c4/ im,jm,iml,jml
      common /c5/ ak9,ak4
      common /c7/ iii,jjj,iiii,jjjj,jn1,jt,jt1
      common /c9/ jn
      common /c18/ pl,rl,ul,vl,wl
```

```
c                free stream levels
      rr       = (1.+(ga-1.)*am*am/2.)**(1./(1.-ga))
      pp       = rr**ga
      uu       = sqrt ((ga-1.)*am*am/2./(1.+(ga-1.)*am*am/2.))
      if (i) 12,13,14
c                read disc field
12    rewind 9
      read (9) p,r,u,v,w,pi,ri,ui,vi,wi
      call pr
      return
c                "on design" i.e.
13    i9       = ifix (1.+hh/dcp)
      i10      = jn + 3
      do 15 i  = 1,im
      do 15 j  = 1,jm
      p(i,j)   = pp
      r(i,j)   = rr
      u(i,j)   = uu
      v(i,j)   = 0.
15    w(i,j)   = 0.
      do 75 ii = 1,imi
      do 75 jj = 1,jmi
      pi(ii,jj) = pp
      ri(ii,jj) = rr
      ui(ii,jj) = uu
      vi(ii,jj) = 0.
75    wi(ii,jj) = 0.
      ak9      = (am*sin((be/180.*pi)+(asin(1./am)-asin(1./amd))*ak4))**2
      p1       = pp*(1.+2.*ga*(ak9-1.)/(ga+1.))
      r1       = rr*(ga+1.)*ak9/(ak9*(ga-1.)+2.)
      q        = sqrt(1.-p1/r1)
      u1       = q*sqrt(1./(1.+tde**2))
      v1       = 0.
      w1       = u1*tde
      do 17 i  = 1,i9
      do 17 j  = 1,i10
      p(i,j)   = p1
      r(i,j)   = r1
      u(i,j)   = u1
      v(i,j)   = v1
17    w(i,j)   = w1
      do 206 i = 1,8
      do 206 j = jt1,i10
      pi(i,j)  = (pp+p1)/2.
      ri(i,j)  = (rr+r1)/2.
      ui(i,j)  = sqrt((1.-pi(i,j)/ri(i,j))/(1.+tde**2))
      vi(i,j)  = 0.
206   wi(i,j)  = ui(i,j)*tde
      call pr
      return
c                new m infinity
14    print 103
      read 104,am
      rewind 9
      read (9) p,r,u,v,w,pi,ri,ui,vi,wi
      rr       = (1.+(ga-1.)*am*am/2.)**(1./(1.-ga))
      pp       = rr**ga
      uu       = sqrt((ga-1.)*am*am/2./(1.+(ga-1.)*am*am/2.))
      do 19 j  = 1,jm
      p(im,j)  = pp
```

```
      r(im,j) = rr
19    u(im,j) = uu
      do 76 jj = 1,jml
      pl(iml,jj) = pp
      rl(iml,jj) = rr
76    ul(iml,jj) = uu
      do 20 i = 1,im
      p(i,jm) = pp
      r(i,jm) = rr
20    u(i,jm) = uu
      do 77 ii = 1,iml
      pl(ii,jml) = pp
      rl(ii,jml) = rr
77    ul(ii,jml) = uu
      call pr
      return
c
103   format ("m=   ")
104   format (f5.3)
c
      end
c
      subroutine integrate(l)
c
      call bc(l)
      call ee(l)
      call fgh(l)
      call prd
      call decode1(l)
      call fgh(l)
      call cor
88    call decode(l)
      return
      end
c
c
      subroutine bc(l)
c             boundary conditions
      dimension p(42,42),r(42,42),u(42,42),v(42,42),w(42,42)
      dimension pl(42,42),rl(42,42),ul(42,42),vl(42,42),wl(42,42)
      dimension etl(42,42),xil(42,42),chil(42),psil(42)
      dimension et(42,42),xi(42,42),chi(42),psi(42)
      common /c1/ p,r,u,v,w
      common /c3/ hh,dcc,dcp,tde,dccl,dcpl
      common /c4/ im,jm,iml,jml
      common /c6/ jjn,a,xiu,tbb,etr
      common /c7/ iii,jjj,iiil,jjjl,jn1,jt,jt1
      common /c8/ xi,et,xil,etl
      common /c9/ jn
      common /c10/ xir,chi,psi,chil,psil
      common /c18/ pl,rl,ul,vl,wl
      if (l.gt.0) go to 78
c             flat surface
      i = 1
      do 79 j = 2,jt
      qq = 1. - pl(i,j)/rl(i,j)
      ul(i,j) = sqrt(qq-(vl(i,j)/cos(atan(tbb)))**2)
79    wl(i,j) = tbb*vl(i,j)
c             centerline
      do 80 i = 1,iiil
```

```
      dxir = (xi(i,2)-xi(i,1))/(xi(i,2)-xi(i+1,2))
      pi(i,1) = pi(i,2)+(pi(i+1,2)-pi(i,2))*dxir
      ri(i,1) = ri(i,2)+(ri(i+1,2)-ri(i,2))*dxir
      ui(i,1) = ui(i,2)+(ui(i+1,2)-ui(i,2))*dxir
      vi(i,1) =-vi(i,2)-(vi(i+1,2)-vi(i,2))*dxir
80    wi(i,1) = wi(i,2)+(wi(i+1,2)-wi(i,2))*dxir
      return
c
c                compression surface
78    i       = 1
      do 30 j = 2,jt
      qq      = 1. - p(i,j)/r(i,j)
      b       = hh*tde*v(i,j)
      d       = (1.+hh**2)*(v(i,j)**2) - qq
      u(i,j)  = (-b+sqrt(b**2-a*d))/a
30    w(i,j)  = tde*u(i,j)+hh*v(i,j)
c                centerline
      do 31 i =1,iii
      dxir    = (xi(i,2)-xi(i,1))/(xi(i+1,2)-xi(i,2))
      p(i,1)  = p(i,2) - (p(i+1,2)-p(i,2))*dxir
      r(i,1)  = r(i,2) - (r(i+1,2)-r(i,2))*dxir
      u(i,1)  = u(i,2) - (u(i+1,2)-u(i,2))*dxir
      v(i,1)  =-v(i,2) + (v(i+1,2)-v(i,2))*dxir
31    w(i,1)  = w(i,2) - (w(i+1,2)-w(i,2))*dxir
      return
      end
c
c
      subroutine decode1(l)
      dimension p(42,42),r(42,42),u(42,42),v(42,42),w(42,42)
      dimension pi(42,42),ri(42,42),ui(42,42),vi(42,42),wi(42,42)
      dimension e(42,42,4),f(42,42,4),g(42,42,4),h(42,42,4)
      dimension e1(42,42,4)
      common /c1/ p,r,u,v,w
      common /c7/ iii,jjj,iiii,jjjj,jn1,jt,jt1
      common /c11/ ak
      common /c14/ e1,e,f,g,h
      common /c17/ negr,negp
      common /c18/ pi,ri,ui,vi,wi
      common /c20/ ja,jb
      ak11    = 1. - ak
      do 36 i = 1,iii
      do 36   j = ja,jjj
      if (l.gt.0) go to 92
      vi(i,j) = e1(i,j,3)/e1(i,j,1)
      wi(i,j) = e1(i,j,4)/e1(i,j,1)
38    ak9     = e1(i,j,2)/e1(i,j,1)
      ak10   = 4.*ak11*ak*(1.-vi(i,j)**2-wi(i,j)**2)
      t       = ak9**2 - ak10
      if (t.ge.0.) go to 37
      negr    = negr + 1
      e1(i,j,2) = 1.01*e1(i,j,1)*sqrt(ak10)
      go to 38
37    ui(i,j)   = (ak9+sqrt(t))*0.5/ak11
      ri(i,j)   = e1(i,j,1)/ui(i,j)
      pi(i,j)   = ri(i,j)*(1.-ui(i,j)**2-vi(i,j)**2-wi(i,j)**2)
      t         = pi(i,j)
      if (t.gt.0.) go to36
      negp      = negp + 1
      go to 36
```

```
92      v(i,j) = e1(i,j,3)/e1(i,j,1)
        w(i,j) = e1(i,j,4)/e1(i,j,1)
93      ak9    = e1(i,j,2)/e1(i,j,1)
        ak10   = 4.*ak11*ak*(1.-v(i,j)**2-w(i,j)**2)
        t      = ak9**2 - ak10
        if (t.ge.0.) go to 94
        negr   = negr + 1
        e1(i,j,2) = 1.01*e1(i,j,1)*sqrt(ak10)
        go to 93
94      u(i,j) = (ak9+sqrt(t))*0.5/ak11
        r(i,j) = e1(i,j,1)/u(i,j)
        p(i,j) = r(i,j)*(1.-u(i,j)**2-v(i,j)**2-w(i,j)**2)
        t      = p(i,j)
        if (t.gt.0.) go to 36
        negp   = negp + 1
36      continue
        return
        end
c
c
        subroutine decode(l)
        dimension p(42,42),r(42,42),u(42,42),v(42,42),w(42,42)
        dimension pl(42,42),rl(42,42),ul(42,42),vl(42,42),wl(42,42)
        dimension e(42,42,4),f(42,42,4),g(42,42,4),h(42,42,4)
        dimension e1(42,42,4)
        common /c1/ p,r,u,v,w
        common /c7/ iii,jjj,iiii,jjjj,jn1,jt,jt1
        common /c11/ ak
        common /c14/ e1,e,f,g,h
        common /c17/ negr,negp
        common /c18/ pl,rl,ul,vl,wl
        ak11    = 1. - ak
        common /c20/ ja,jb
        do 36 i = 1,iii
        do 36   j = ja,jjj
        if (l.gt.0) go to 92
        vl(i,j) = e(i,j,3)/e(i,j,1)
        wl(i,j) = e(i,j,4)/e(i,j,1)
38      ak9     = e(i,j,2)/e(i,j,1)
        ak10    = 4.*ak11*ak*(1.-vl(i,j)**2-wl(i,j)**2)
        t       = ak9**2 - ak10
        if (t.ge.0.) go to 37
        negr    = negr + 1
        e(i,j,2) = 1.01*e(i,j,1)*sqrt(ak10)
        go to 38
37      ul(i,j)   = (ak9+sqrt(t))*0.5/ak11
        rl(i,j)   = e(i,j,1)/ul(i,j)
        pl(i,j)   = rl(i,j)*(1.-ul(i,j)**2-vl(i,j)**2-wl(i,j)**2)
        t         = pl(i,j)
        if (t.gt.0.) go to36
        negp      = negp + 1
        go to 36
92      v(i,j) = e(i,j,3)/e(i,j,1)
        w(i,j) = e(i,j,4)/e(i,j,1)
93      ak9    = e(i,j,2)/e(i,j,1)
        ak10   = 4.*ak11*ak*(1.-v(i,j)**2-w(i,j)**2)
        t      = ak9**2 - ak10
        if (t.ge.0.) go to 94
        negr   = negr + 1
        e(i,j,2) = 1.01*e(i,j,1)*sqrt(ak10)
```

```
      go to 93
 94   u(i,j) = (ak9+sqrt(t))*0.5/ak11
      r(i,j) = e(i,j,1)/u(i,j)
      p(i,j) = r(i,j)*(1.-u(i,j)**2-v(i,j)**2-w(i,j)**2)
      t      = p(i,j)
      if (t.gt.0.) go to 36
      negp   = negp + 1
 36   continue
      return
      end
c
c
c
      subroutine prd
      dimension e(42,42,4),f(42,42,4),g(42,42,4),h(42,42,4)
      dimension e1(42,42,4)
      common /c6/ jjn,a,xiu,tbb,etr
      common /c7/ iii,jjj,iiii,jjjj,jn1,jt,jt1
      common /c9/ jn
      common /c14/ e1,e,f,g,h
      common /c15/ dz,al,bet,z
      common /c20/ ja,jb
      do 98 k = 1,4
      do 27 j = ja,jjj
      do 27 i = 2,iii
      e1(i,j,k) = e(i,j,k) - al*(f(i,j,k)-f(i,j-1,k))-bet*(g(i+1,j,k)
     &                  -g(i,j,k)) - (h(i,j,k)+2.*e(i,j,k))*dz
 27   continue
      do 97 j = ja,jt
 97   e1(1,j,k) = e(1,j,k) - al*(f(1,j,k)-f(1,j-1,k))-bet*(g(2,j,k)
     &                  -g(1,j,k)) - (h(1,j,k)+2.*e(1,j,k))*dz
      do 201 j = jt1,jjj
 201  e1(1,j,k) = e(1,j,k)
 98   continue
      return
      end
c
      subroutine cor
      dimension e(42,42,4),f(42,42,4),g(42,42,4),h(42,42,4)
      dimension e1(42,42,4)
      common /c6/ jjn,a,xiu,tbb,etr
      common /c7/ iii,jjj,iiii,jjjj,jn1,jt,jt1
      common /c14/ e1,e,f,g,h
      common /c15/ dz,al,bet,z
      common /c20/ ja,jb
      do 200 k = 1,4
      do 28 j = ja,jjj
      do 28 i= 2,iii
      e(i,j,k)  = 0.5*(e(i,j,k)+e1(i,j,k)-al*(f(i,j+1,k)-f(i,j,k))
     &                  -bet*(g(i,j,k)-g(i-1,j,k))-(h(i,j,k)+2.*e1(i,j,k))*dz)
 28   continue
      do 99 j = ja,jt
      e(1,j,k)  = 0.5*(e(1,j,k)+e1(1,j,k)-al*(f(1,j+1,k)-f(1,j,k))
     &                  -bet*(g(2,j,k)-g(1,j,k))-(h(1,j,k)+2.*e1(1,j,k))*dz)
 99   continue
 200  continue
      return
      end
c
      subroutine ee(l)
```

```
      dimension p(42,42),r(42,42),u(42,42),v(42,42),w(42,42)
      dimension pl(42,42),rl(42,42),ul(42,42),vl(42,42),wl(42,42)
      dimension e(42,42,4),f(42,42,4),g(42,42,4),h(42,42,4)
      dimension e1(42,42,4)
      common /c1/ p,r,u,v,w
      common /c4/ im,jm,iml,jml
      common /c11/ ak
      common /c14/ e1,e,f,g,h
      common /c18/ pl,rl,ul,vl,wl
      common /c20/ ja,jb
      do 90 i = 1,im
      do 90 j = jb,jm
      if (i.gt.0) go to 91
      e(i,j,1) = rl(i,j)*ul(i,j)
      e(i,j,2) = pl(i,j)*ak + e(i,j,1)*ul(i,j)
      e(i,j,3) = e(i,j,1)*vl(i,j)
      e(i,j,4) = e(i,j,1)*wl(i,j)
      go to 90
91    e(i,j,1) = r(i,j)*u(i,j)
      e(i,j,2) = p(i,j)*ak + e(i,j,1)*u(i,j)
      e(i,j,3) = e(i,j,1)*v(i,j)
      e(i,j,4) = e(i,j,1)*w(i,j)
90    continue
      return
      end
c
      subroutine fgh(i)
      dimension p(42,42),r(42,42),u(42,42),v(42,42),w(42,42)
      dimension pl(42,42),rl(42,42),ul(42,42),vl(42,42),wl(42,42)
      dimension et(42,42),xi(42,42),chi(42),psi(42)
      dimension etl(42,42),xil(42,42),chil(42),psil(42)
      dimension e(42,42,4),f(42,42,4),g(42,42,4),h(42,42,4)
      dimension e1(42,42,4)
      dimension fb(4),gb(4)
      common /c1/ p,r,u,v,w
      common /c4/ im,jm,iml,jml
      common /c6/ jjn,a,xiu,tbb,etr
      common /c8/ xi,et,xil,etl
      common /c10/ xir,chi,psi,chil,psil
      common /c11/ ak
      common /c13/ s
      common /c14/ e1,e,f,g,h
      common /c16/ delchib,delpsib,delchil
      common /c18/ pl,rl,ul,vl,wl
      common /c20/ ja,jb
      do 34 i = 1,im
      do 34 j = jb,jm
      if (i.gt.0) go to 86
      vv      = vl(i,j)/s - etl(i,j)*ul(i,j)
      ww      = wl(i,j)/s - xil(i,j)*ul(i,j)
      fb(1)   = vv*rl(i,j)
      fb(2)   = vv*rl(i,j)*ul(i,j) - ak*pl(i,j)*etl(i,j)
      fb(3)   = vv*rl(i,j)*vl(i,j) + ak*pl(i,j)/s
      fb(4)   = vv*rl(i,j)*wl(i,j)
      gb(1)   = ww*rl(i,j)
      gb(2)   = ww*rl(i,j)*ul(i,j) - ak*pl(i,j)*xil(i,j)
      gb(3)   = ww*rl(i,j)*vl(i,j)
      gb(4)   = ww*rl(i,j)*wl(i,j) + ak*pl(i,j)/s
      do 85 k = 1,4
      f(i,j,k)  = (chil(j)*gb(k)+fb(k))*chil(j)/etl(i,j)
```

```
      g(i,j,k)  = gb(k) + tbb*fb(k)
      h(i,j,k)  = (tbb*fb(k)+(1.+2.*chi(j))*tbb)*gb(k))/(psi(i)-xiu)
      f(i,j,k)  = f(i,j,k)/delchi
85    g(i,j,k)  = g(i,j,k)/psi(imi)
      go to 34
86    vv        = v(i,j)/s - et(i,j)*u(i,j)
      ww        = w(i,j)/s - xi(i,j)*u(i,j)
      fb(1)     = vv*r(i,j)
      fb(2)     = vv*r(i,j)*u(i,j) - ak*p(i,j)*et(i,j)
      fb(3)     = vv*r(i,j)*v(i,j) + ak*p(i,j)/s
      fb(4)     = vv*r(i,j)*w(i,j)
      gb(1)     = ww*r(i,j)
      gb(2)     = ww*r(i,j)*u(i,j) - ak*p(i,j)*xi(i,j)
      gb(3)     = ww*r(i,j)*v(i,j)
      gb(4)     = ww*r(i,j)*w(i,j) + ak*p(i,j)/s
      do 35 k = 1,4
      f(i,j,k)  = (chi(j)*gb(k) + fb(k))*chi(j)/et(i,j)
      g(i,j,k)  = (psi(i)*fb(k) + gb(k))*psi(i)/(xi(i,j)-xir)
      h(i,j,k)  = (psi(i)*f(i,j,k)+chi(j)*g(i,j,k))/(1.-psi(i)*chi(j))
     &                -fb(k)*psi(i)/(xi(i,j)-xir)
     &                -gb(k)*chi(j)/et(i,j)
      f(i,j,k)  = f(i,j,k)/delchib
      g(i,j,k)  = g(i,j,k)/delpsib
35    continue
34    continue
      return
      end
c
c
c
      subroutine pr
      dimension p(42,42),r(42,42),u(42,42),v(42,42),w(42,42)
      dimension et(42,42),xi(42,42)
      dimension pi(42,42),ri(42,42),ui(42,42),vi(42,42),wi(42,42)
      dimension eti(42,42),xii(42,42)
      common /c1/ p,r,u,v,w
      common /c4/ im,jm,imi,jmi
      common /c7/ iii,jjj,iiii,jjjj,jn1,jt,jt1
      common /c9/ jn
      common /c18/ pi,ri,ui,vi,wi
c                partial output
      print 110
      print 113, (p(1,j),j=1,jt1,2)
      print 114, (pi(1,ji), ji=1,jt1,2)
      print 148, (ri(1,ji), ji=1,jt1,2)
      print 149, (vi(1,ji), ji=1,jt1,2)
      print 110
      print 113, (p(i,2),i=1,im,2)
      print 114, (pi(ii,2), ii=1,imi,2)
      print 110
      do 40 k = 1,3
40    print 113, (p(k,j),j=jt1,jm)
      print 110
      do 63 k = 1,3
63    print 114, (pi(k,ji), ji=jt1,jmi)
      print 110
      return
c
110   format (2x)
113   format ("p ",2x,20f8.4)
```

```
114     format ("pl",2x,20f8.4)
148     format ("rl",2x,20f8.4)
149     format ("vl",2x,20f8.4)
c
        end
c
c
        subroutine prcoord
        dimension et(42,42),xi(42,42)
        dimension etl(42,42),xil(42,42)
        common /c4/ im,jm,iml,jml
        common /c8/ xi,et,xil,etl
        common /c9/ jn
c                partial coord output
        jn1    = jn +  1
        print 121
        print 111, (et(1,j),j=1,jn1,4)
        print 121
        return
c
111     format ("et",2x,20f8.4)
121     format (2x)
c
        end
c
c
        subroutine prfin
        dimension p(42,42),r(42,42),u(42,42),v(42,42),w(42,42)
        dimension et(42,42),xi(42,42),chi(42),psi(42)
        dimension pl(42,42),rl(42,42),ul(42,42),vl(42,42),wl(42,42)
        dimension etl(42,42),xil(42,42),chil(42),psil(42)
        dimension etb(42),xib(42)
        dimension p2(42,42),p3(42,42)
        common /c1/ p,r,u,v,w
        common /c2/ ga,am,be,pi,amd
        common /c3/ hh,dcc,dcp,tde,dccl,dcpl
        common /c4/ im,jm,iml,jml
        common /c6/ jjn,a,xiu,tbb,etr
        common /c7/ iii,jjj,iiil,jjjl,jn1,jt,jt1
        common /c8/ xi,et,xil,etl
        common /c9/ jn
        common /c10/ xir,chi,psi,chil,psil
        common /c18/ pl,rl,ul,vl,wl
c                coordinates
        print 125
        do 52 i=1,im,2
52      print 126, (et(i,j),j=1,jm,2)
        print 125
        do 53 i=1,im,2
53      print 127, (xi(i,j),j=1,jm,2)
        do 46 j = jn,jm
        etb(j)   =  (chi(j)*xiu)/(1.+chi(j)*tbb)
46      xib(j)   =  tbb*etb(j)
        print 125
        print 128, (etb(j),j=jn,jm)
        print 129, (xib(j),j=jn,jm)
        print 125
        do 64 ii =1,iml,2
64      print 138, (etl(ii,jj),jj=1,jml,2)
        print 125
```

```
      do 65 il = 1,iml,2
65    print 139, (xil(il,jl),jl=1,jml,2)
      print 125
c                field
      ak9  = 2./(ga*(am**2))
      do 47 j=1,jm
      do 47 i=1,im
47    p2(i,j) = p(i,j)/p(im,jm)
      do 95 i= 1,im
95    print 130, (p2(i,j),j=1,14)
      print 125
      do 202 i = 1,im
202   print 130, (p2(i,j),j=15,28)
      print 125
      do 54 i=1,2
54    print 130, (p2(i,j),j=29,jm)
      print 125
      do 55 i=1,im,6
55    print 131, ( r(i,j),j=1,jm,4)
      print 125
      do 56 i=1,im,6
56    print 132, ( u(i,j),j=1,jm,4)
      print 125
      do 57 i=1,im,6
57    print 133, ( v(i,j),j=1,jm,4)
      print 125
      do 58 i=1,im,6
58    print 134, ( w(i,j),j=1,jm,4)
      print 125
      jjn4  =  jn - 4
      jjn2  =  jn + 2
      print 131, ( r(1,j),j=jjn4,jjn2)
      print 132, ( u(1,j),j=jjn4,jjn2)
      print 133, ( v(1,j),j=jjn4,jjn2)
      print 134, ( w(1,j),j=jjn4,jjn2)
      print 125
      do 66 jl = 1,jml
      do 66 il = 1,iml
66    p3(il,jl) = pl(il,jl)/pl(iml,jml)
      do 96 il = 1,iml
96    print 140, (p3(il,jl),jl=1,14)
      print 125
      do 203 il = 1,iml
203   print 140, (p3(il,jl),jl=15,28)
      print 125
      do 67 il = 1,2
67    print 140, (p3(il,jl),jl=29,jml)
      print 125
      do 68 il = 1,iml,6
68    print 141, (rl(il,jl),jl=1,jml,4)
      print 125
      do 69 il = 1,iml,6
69    print 142, (ul(il,jl),jl=1,jml,4)
      print 125
      do 70 il = 1,iml,6
70    print 143, (vl(il,jl),jl=1,jml,4)
      print 125
      do 71 il = 1,iml,6
71    print 144, (wl(il,jl),jl=1,jml,4)
      print 125
```

```
c
      clp   = 0.
      clpl  = 0.
      jjn3  = jt - 1
      do 60 j=2,jjn3
60    clp   = clp + (p2(1,j+1)+p2(1,j))*(et(1,j+1)-et(1,j))/2.
      clp   = clp + (p2(1,2)+p2(1,1))*(et(1,2)-et(1,1))/4.
      clp   = clp + p2(1,jjn)*(1.-et(1,jjn))
      do 74 jl = 2,jjn3
74    clpl = clpl + (p3(1,jl+1)+p3(1,jl))*(etl(1,jl+1)-etl(1,jl))/2.
      clpl = clpl +(p3(1,2)+p3(1,1))*(etl(1,2)-etl(1,1))/4.
      clpl = clpl + p3(1,jjn)*(1.-etl(1,jjn))
      cl   = ak9*(clp-clpl)
      cd   = ak9*clp*tde
      cld  = cl/cd
      print 137, cl,cd,cld
      print 125
      do 48 j = 1,jm
      do 48 i = 1,im
48    p2(i,j) = ak9*(p2(i,j)-1.)
      print 136, (p2(1,j),j=1,14)
      print 136,  (p2(1,j),j=15,28)
      print 136, (p2(1,j),j=29,jm)
      print 125
      do 72 jl = 1,jml
      do 72 il = 1,iml
72    p3(il,jl) = ak9*(p3(il,jl)-1.)
      print 145, (p3(1,j),j=1,12)
      print 145, (p3(1,j),j=13,jm)
      print 125
      return
c
125   format (2x)
126   format ("et",2x,16f8.4)
127   format ("xi",2x,16f8.4)
128   format ("etb",x,16f8.4)
129   format ("xib",x,16f8.4)
130   format ("p",3x,16f8.4)
131   format ("r",3x,16f8.4)
132   format ("u",3x,16f8.4)
133   format ("v",3x,16f8.4)
134   format ("w",3x,16f8.4)
136   format ("cp",2x,16f8.4)
137   format ("cl =",f8.4,4x,"cd =",f8.4,4x,"cl/cd =",f8.4)
138   format ("etl",x,16f8.4)
139   format ("xil",x,16f8.4)
140   format ("pl",2x,16f8.4)
141   format ("rl",2x,16f8.4)
142   format ("ul",2x,16f8.4)
143   format ("vl",2x,16f8.4)
144   format ("wl",2x,16f8.4)
145   format ("cpl",x,16f8.4)
c
      end
c
c
      subroutine mod(l)
      dimension p(42,42),r(42,42),u(42,42),v(42,42),w(42,42)
      dimension pl(42,42),rl(42,42),ul(42,42),vl(42,42),wl(42,42)
      dimension et(42,42),xi(42,42),chi(42),psi(42)
```

```
      dimension etl(42,42),xil(42,42),chil(42),psil(42)
      common /c1/ p,r,u,v,w
      common /c3/ hh,dcc,dcp,tde,dccl,dcpl
      common /c4/ im,jm,iml,jml
      common /c6/ jjn,a,xiu,tbb,etr
      common /c7/ iii,jjj,iiil,jjjl,jn1,jt,jt1
      common /c8/ xi,et,xil,etl
      common /c9/ jn
      common /c10/ xir,chi,psi,chil,psil
      common /c18/ pl,rl,ul,vl,wl
      common /c13/ s
      if (l.gt.0) go to 83
      do 82 jl = jt1,jjj
      psis  = tde*(1.+(tbb-xiu)*chil(jl))/(s*(1.+hh*chil(jl)))
      di    = 1. + (psis/dcpl)
      il    = ifix(di)
      dxir  = (xil(il,jl)-xi(1,jl))/(xil(il,jl)-xil(il+1,jl))
      p(1,jl)  = pl(il,jl)+(pl(il+1,jl)-pl(il,jl))*dxir
      r(1,jl)  = rl(il,jl)+(rl(il+1,jl)-rl(il,jl))*dxir
      u(1,jl)  = ul(il,jl)+(ul(il+1,jl)-ul(il,jl))*dxir
      v(1,jl)  = vl(il,jl)+(vl(il+1,jl)-vl(il,jl))*dxir
82    w(1,jl)  = wl(il,jl)+(wl(il+1,jl)-wl(il,jl))*dxir
      return
c
83    do 81 j = jt1,jjj
      psis  = ((xiu-xir)*tbb*chi(j)-xir)/((etr*tbb-xiu)*chi(j)+etr)
      di    = 1. + (psis+hh)/dc
      i     = ifix(di)
      dxir  = (xil(1,j)-xi(i,j))/(xi(i+1,j)-xi(i,j))
      pl(1,j) = p(i,j)+(p(i+1,j)-p(i,j))*dxir
      rl(1,j) = r(i,j)+(r(i+1,j)-r(i,j))*dxir
      ul(1,j) = u(i,j)+(u(i+1,j)-u(i,j))*dxir
      vl(1,j) = v(i,j)+(v(i+1,j)-v(i,j))*dxir
81    wl(1,j) = w(i,j)+(w(i+1,j)-w(i,j))*dxir
      return
c
      end
```

Table 1

Normal Section Angle Dependence on Span and Shock Angle for Design Mach Numbers 2 and 4

s*	δ_N/δ				
	β_D^o = 45	60	30	45	60
	δ/β_D = 0.328	0.374	0.593	0.663	0.629
	(δ^o = 14.744	22.411	17.784	29.846	37.758)
	(M_D = 2.	2.	4.	4.	4.)
0	0.	0.	0.	0.	0.
0.2	0.3749	0.1819	1.1225	0.6490	0.2751
0.4	0.6638	0.3495	1.5164	1.0023	0.5075
0.6	0.8458	0.4931	1.4753	1.1367	0.6800
0.8	0.9463	0.6092	1.3820	1.1718	0.7983
1.0	0.9974	0.6995	1.2995	1.1683	0.8761
2.0	1.0327	0.9086	1.1052	1.0862	0.9998
4.0	1.0133	0.9804	1.0292	1.0272	1.0095
10.0	1.0024	0.9973	1.0048	1.0047	1.0023
∞	1.	1.	1.	1.	1.

Table 2

On-Design Normal Section Shock Angle

s*	β_N^o		
	$\beta_D^o = 30$	45	60
0	90.	90.	90.
0.2	73.532	82.029	86.712
0.4	60.449	74.775	83.540
0.6	51.388	68.666	80.580
0.8	45.392	63.786	77.898
1.0	41.410	60.000	75.523
2.0	33.690	50.769	67.792
4.0	31.003	46.686	62.688
10.0	30.165	45.284	60.484
20.0	30.049	45.071	60.123
∞	30.	45.	60.

Table 3

Off-Design Attached Shock Conditions

s*	β_D^o	δ^o	δ_N^o	M_∞	M_N	β_N^o	β_D^o	$\frac{p_2}{p_\infty}$	C_{p_2}	$(\tau_D-\tau)^o$	$(\frac{v}{q})_2$
1	35	5.746	6.382	2.0	1.547	47.855	35.	1.3686	.1316	0	0
				1.9	1.470	51.367	37.228	1.3713	.1469	0.41	0.006
				1.8	1.393	55.915	40.129	1.3851	.1698	1.04	0.016
				1.684*	1.303	65.004	46.064	1.4597	.2316	2.82	0.042
2	35	5.746	6.090	2.0	1.809	39.363	35.	1.3686	.1316	0	0
				1.8	1.628	44.500	39.388	1.3520	.1552	0.37	0.006
				1.6	1.447	52.067	45.893	1.3530	.1970	1.04	0.017
				1.428*	1.292	65.310	57.556	1.441	.3089	3.00	0.050
2	45	14.744	15.225	2.0	1.826	50.769	45.	2.1667	.4167		
				1.9	1.734	54.640	48.165	2.1675	.4620		
				1.797*	1.640	61.494	53.791	2.2577	.5564		
1	35	22.181	28.102	4	3.094	47.855	35.	5.975	.4442		
				3.5	2.708	52.410	37.890	5.204	.4903		
				3.057*	2.365	62.284	44.261	4.947	.6034		
2	35	22.181	24.436	4.	3.617	39.363	35.	5.978	.4442		
				3.5	3.165	42.056	37.300	5.078	.4756		
				3.0	2.783	46.285	40.918	4.320	.5269		
				2.5	2.261	54.468	47.975	3.783	.6361		
				2.326*	2.106	61.686	54.314	3.834	.7483		

* M_∞ such that $M_{N_2} = 1.0$

Table 4

Case Constants

Attached Algorithm

s*	M_∞	$i_m \times j_m$	$(\eta)_{j_m}$
1	1.9	14 x 14	.85
	1.8		.90
2	2.0	28 x 28	.50
	1.8		.60
	1.6		.65
	1.55	16 x 16	.80
	1.50		.85
1	4.0		.70
	3.5		.80
2	4.0		.40
	3.5	14 x 14	.50
	3.0		.55
	2.5		.75

Detached Algorithm

s*	M_∞	$i_m \times j_m$	η_R	η_M	η_N	ξ_u	ξ_C
1	1.6	24 x 24	1.35	1.30	1.05	2.637	2.276
2	1.7	28 x 24	1.467	1.40	↓	5.882	1.225
	1.5	24 x 24	↓	↓	↓	↓	0.980
	1.4	24 x 24	↓	↓	↓	↓	1.225
	1.35	16 x 24	↓	↓	↓	↓	0.980
	1.3	20 x 24	↓	↓	↓	↓	0.980

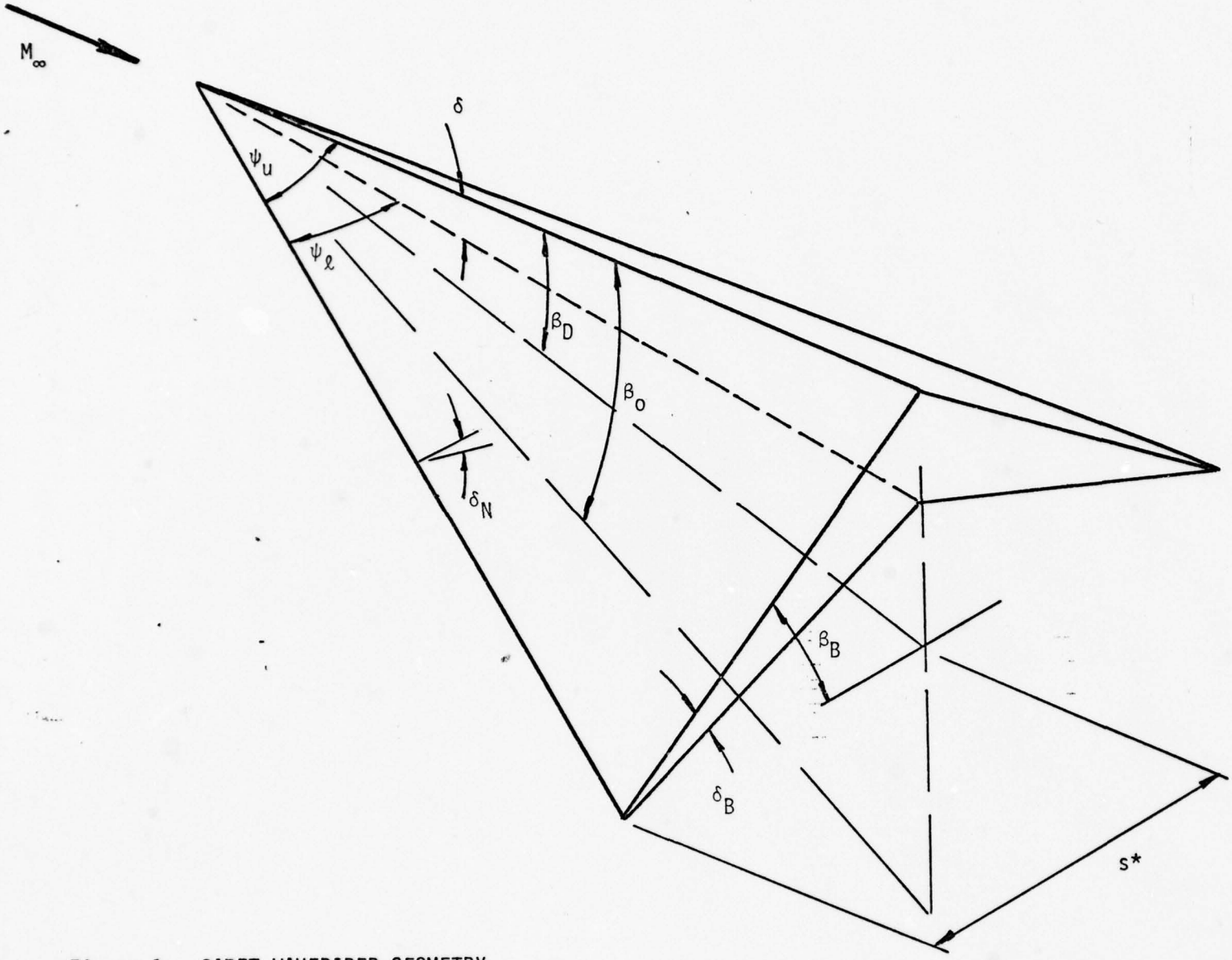


Figure 1. CARET WAVERIDER GEOMETRY

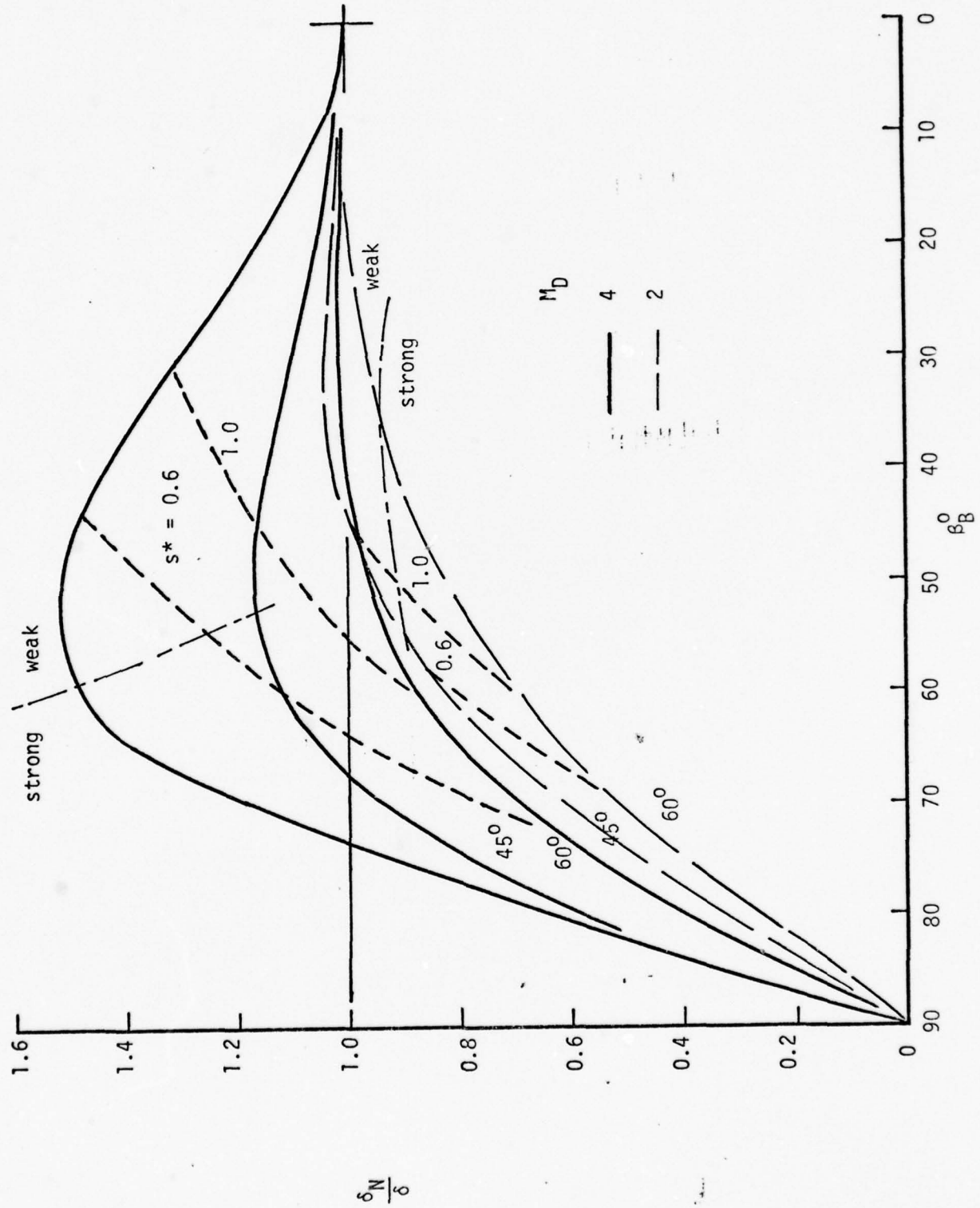


Figure 2. NORMAL SECTION ANGLE VARIATION IN TERMS OF DESIGN CONDITIONS

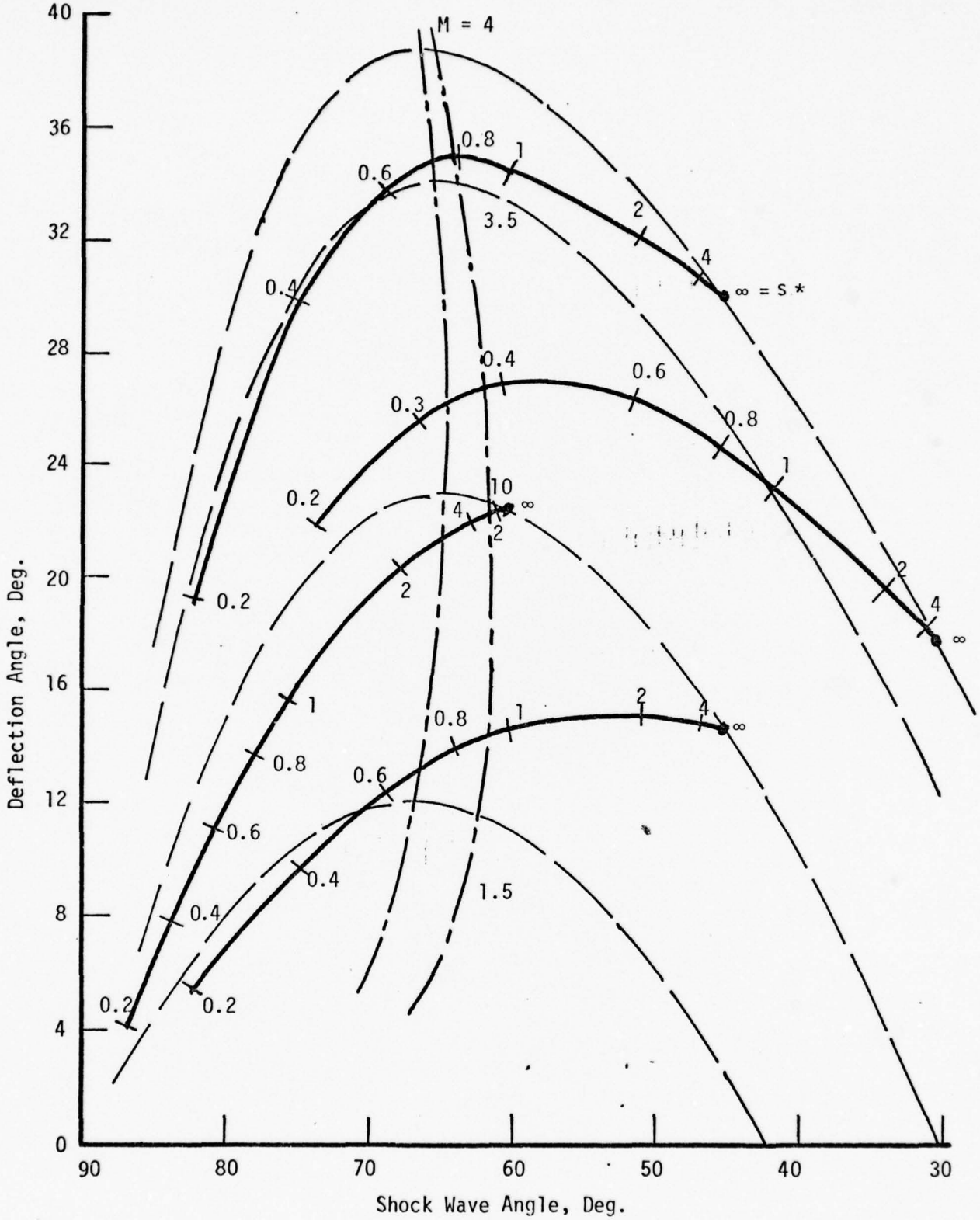


Figure 3. SPAN INFLUENCE ON SIDE EDGE SHOCK OF WEAK AND STRONG TYPE

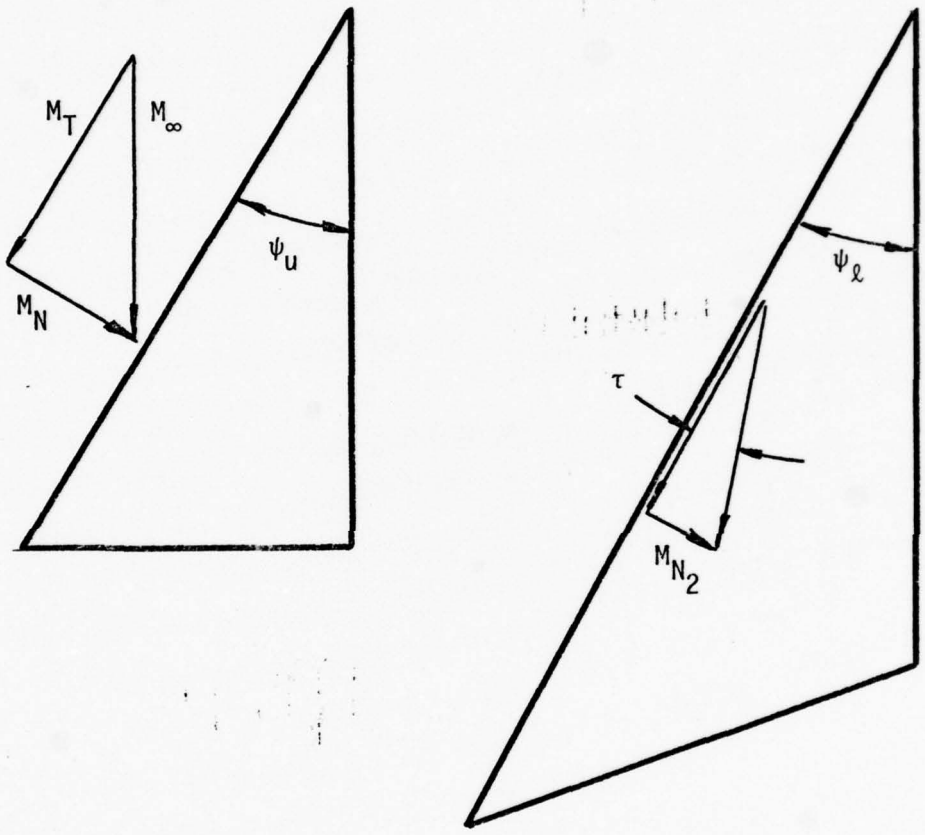
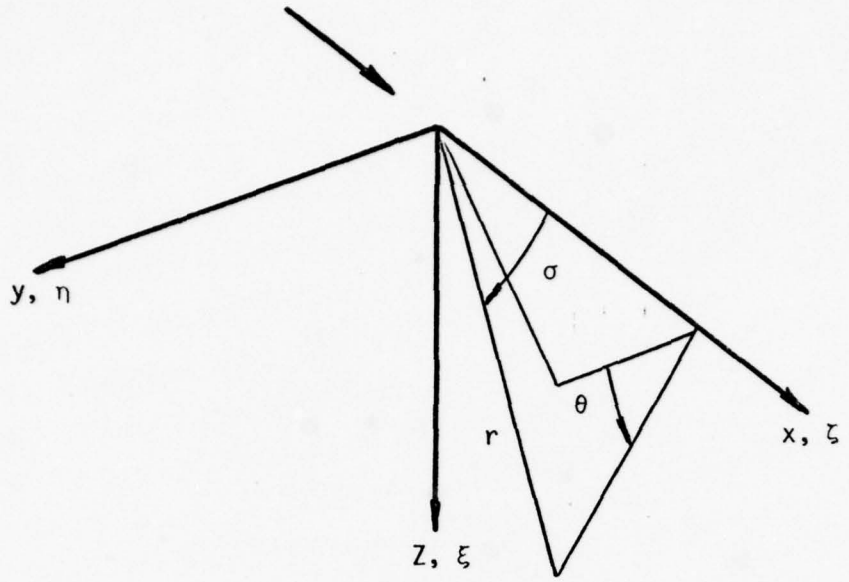


Figure 4. ATTACHED SHOCK FLOW TURNING IN SURFACE PLANE



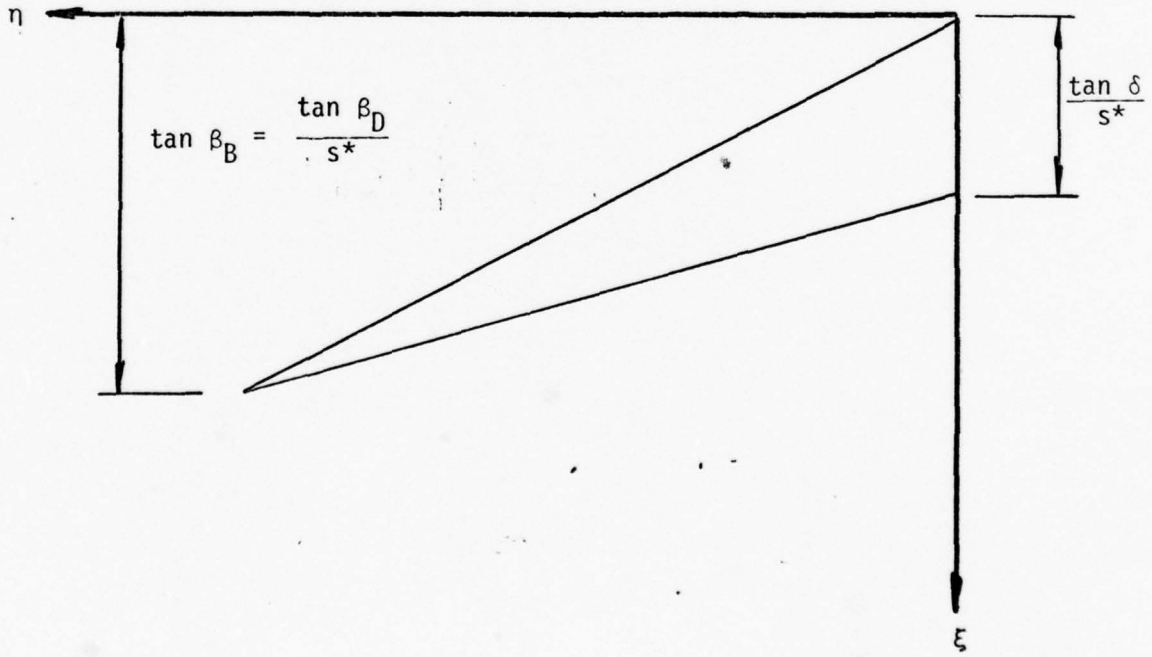


Figure 5. COORDINATE SYSTEM

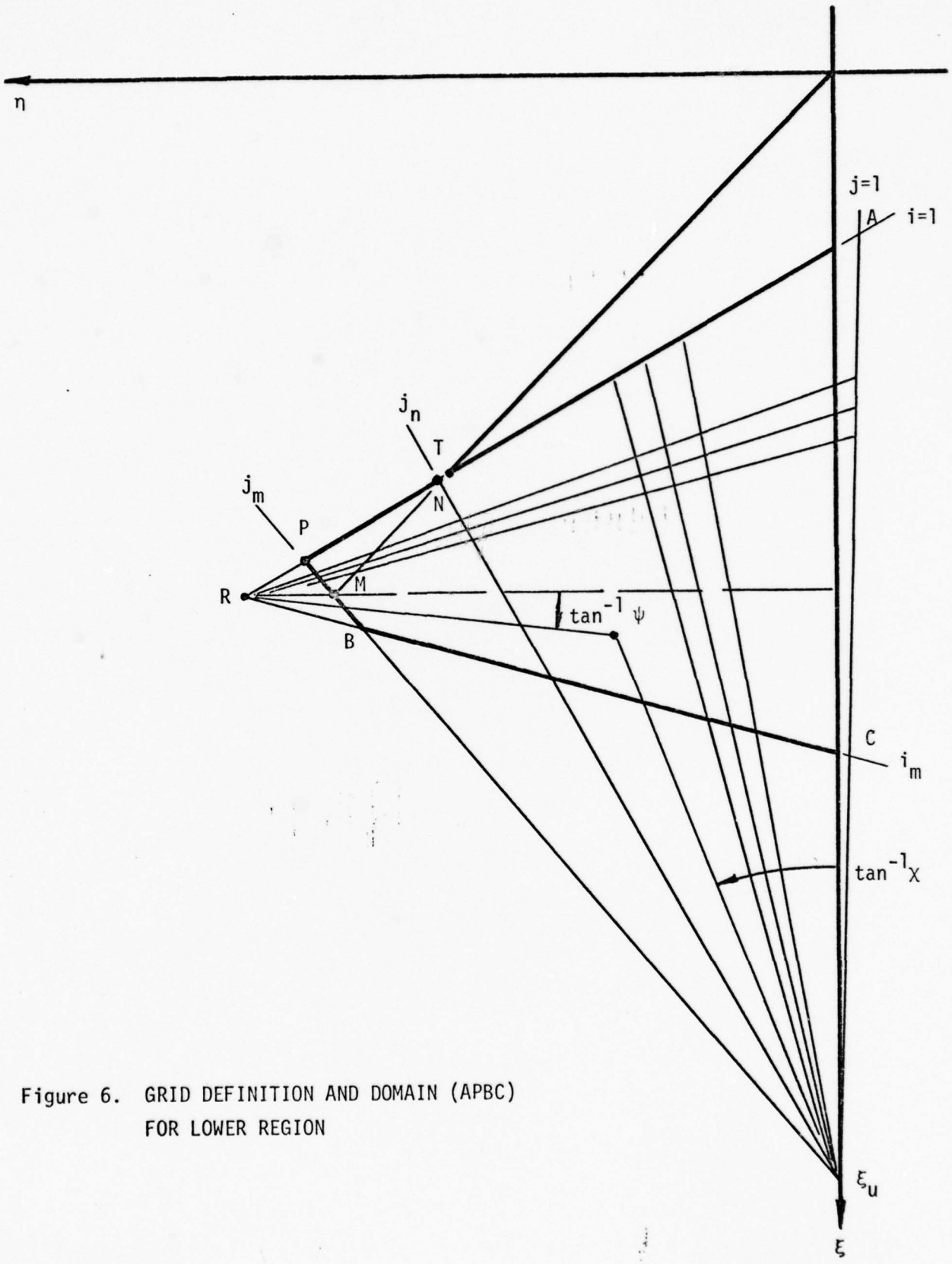
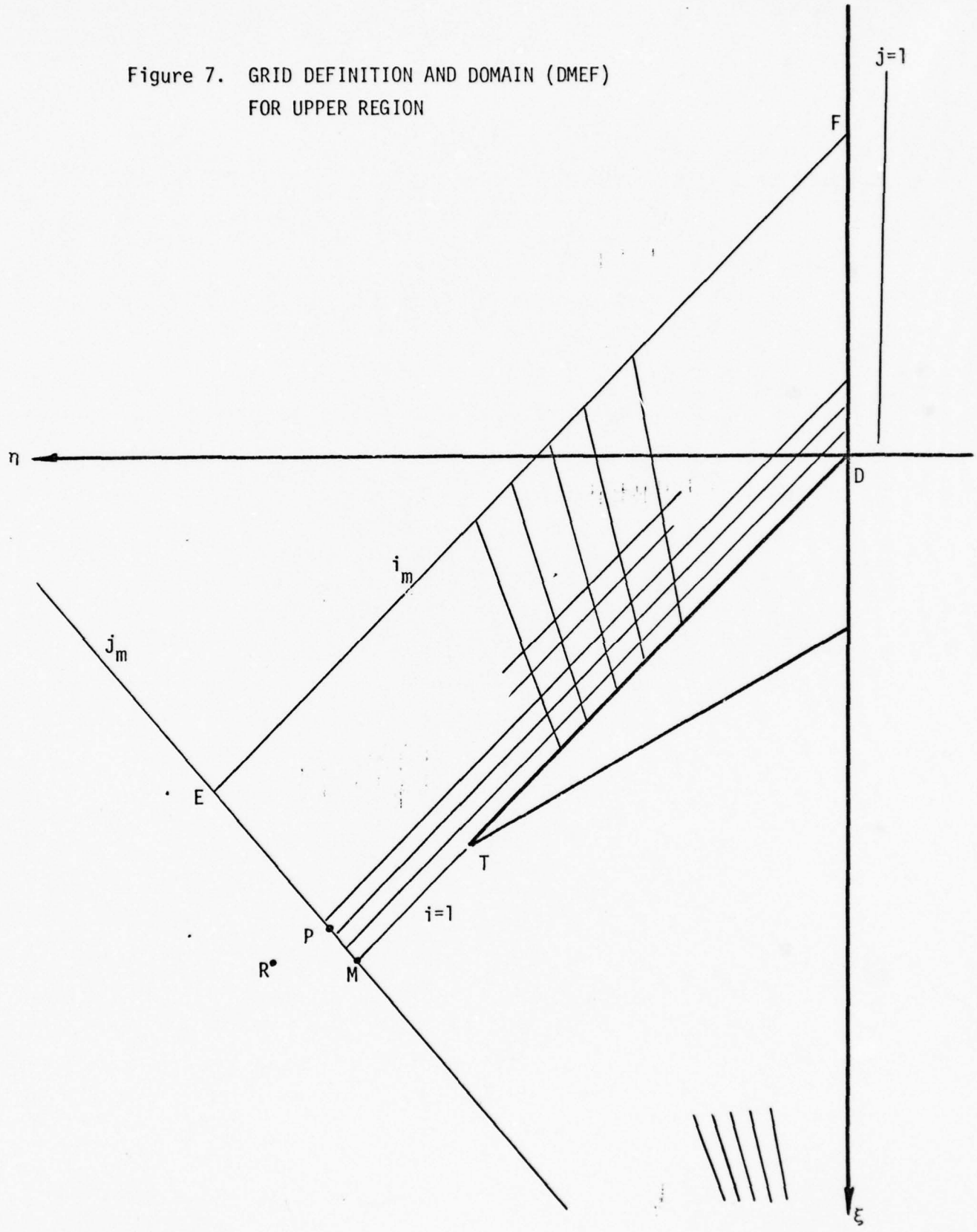


Figure 6. GRID DEFINITION AND DOMAIN (APBC) FOR LOWER REGION

Figure 7. GRID DEFINITION AND DOMAIN (DMEF)
FOR UPPER REGION



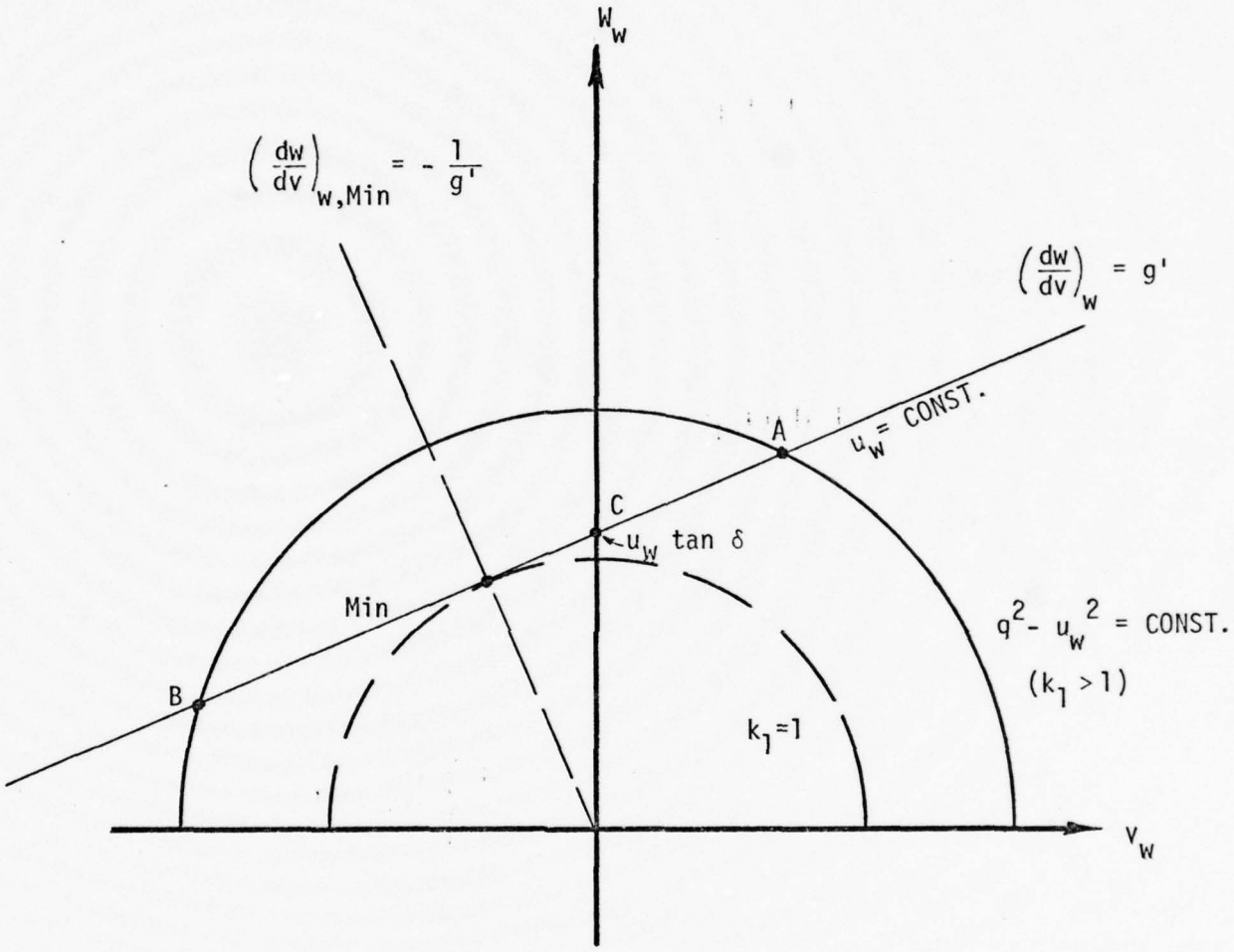


Figure 8. SURFACE BOUNDARY CONDITION CONSTRAINTS

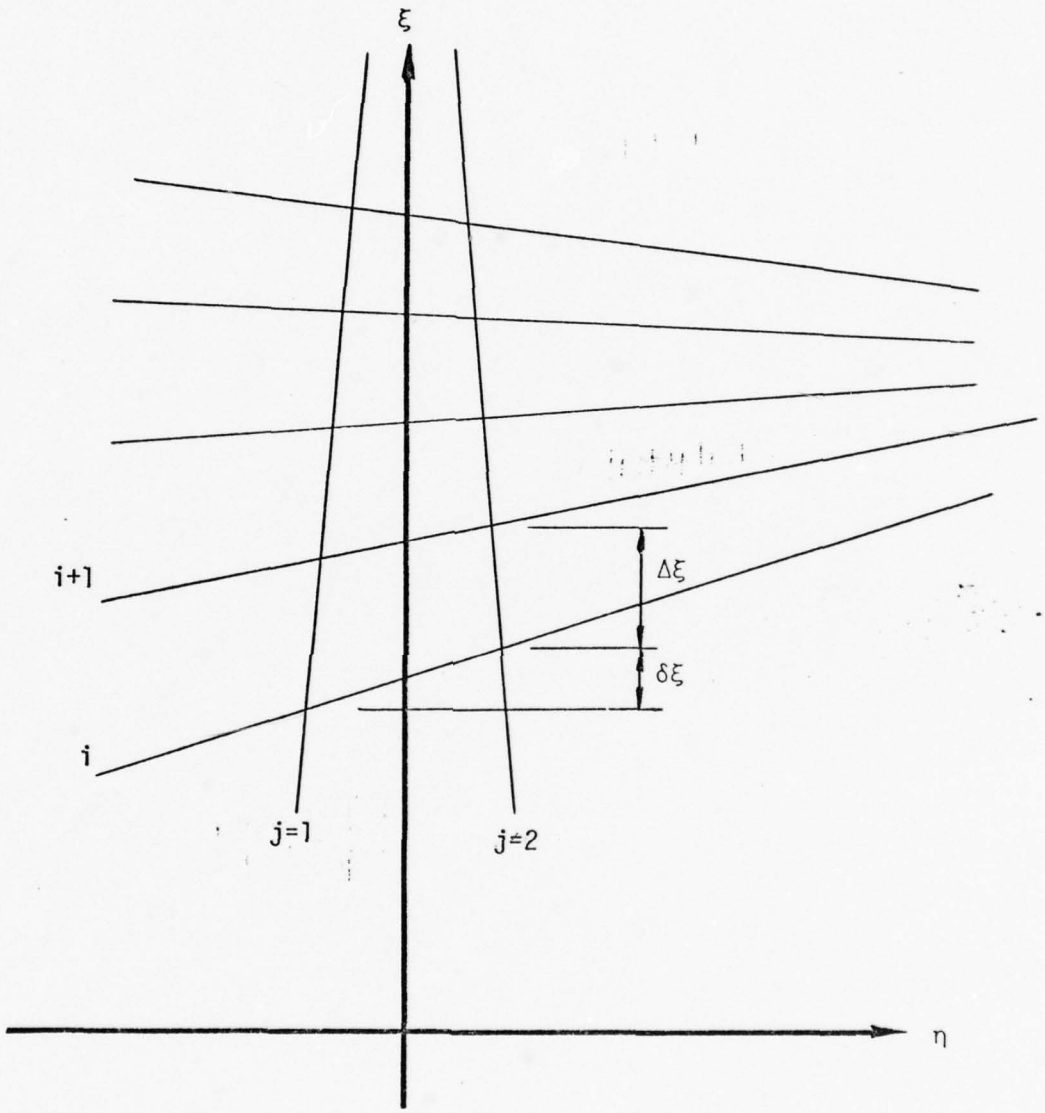


Figure 9. SYMMETRY PLANE BOUNDARY CONDITION INTERPOLATION

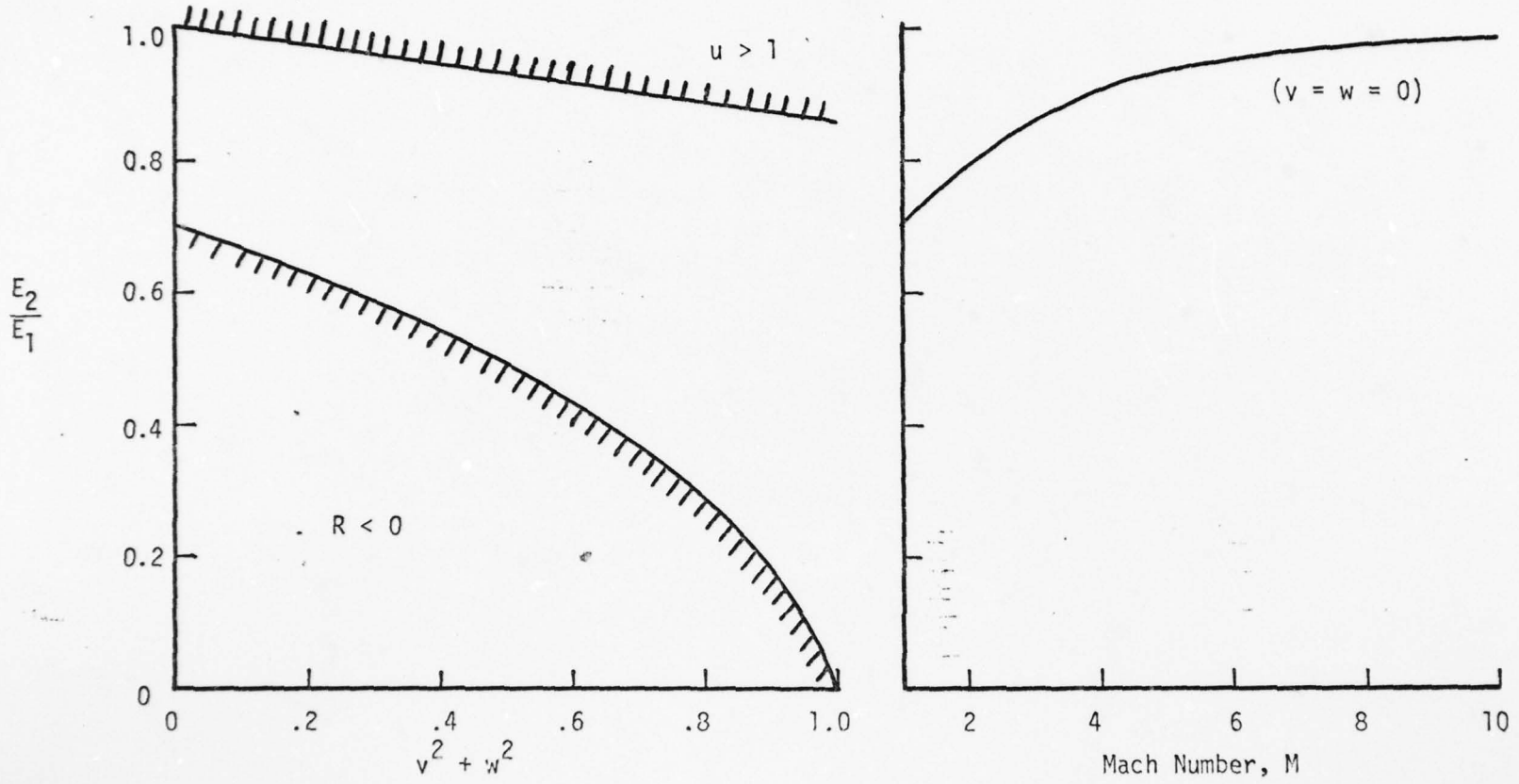


Figure 10. DECODING CONSTRAINT FOR PHYSICALLY INTERPRETABLE STATE VECTOR COMPONENTS

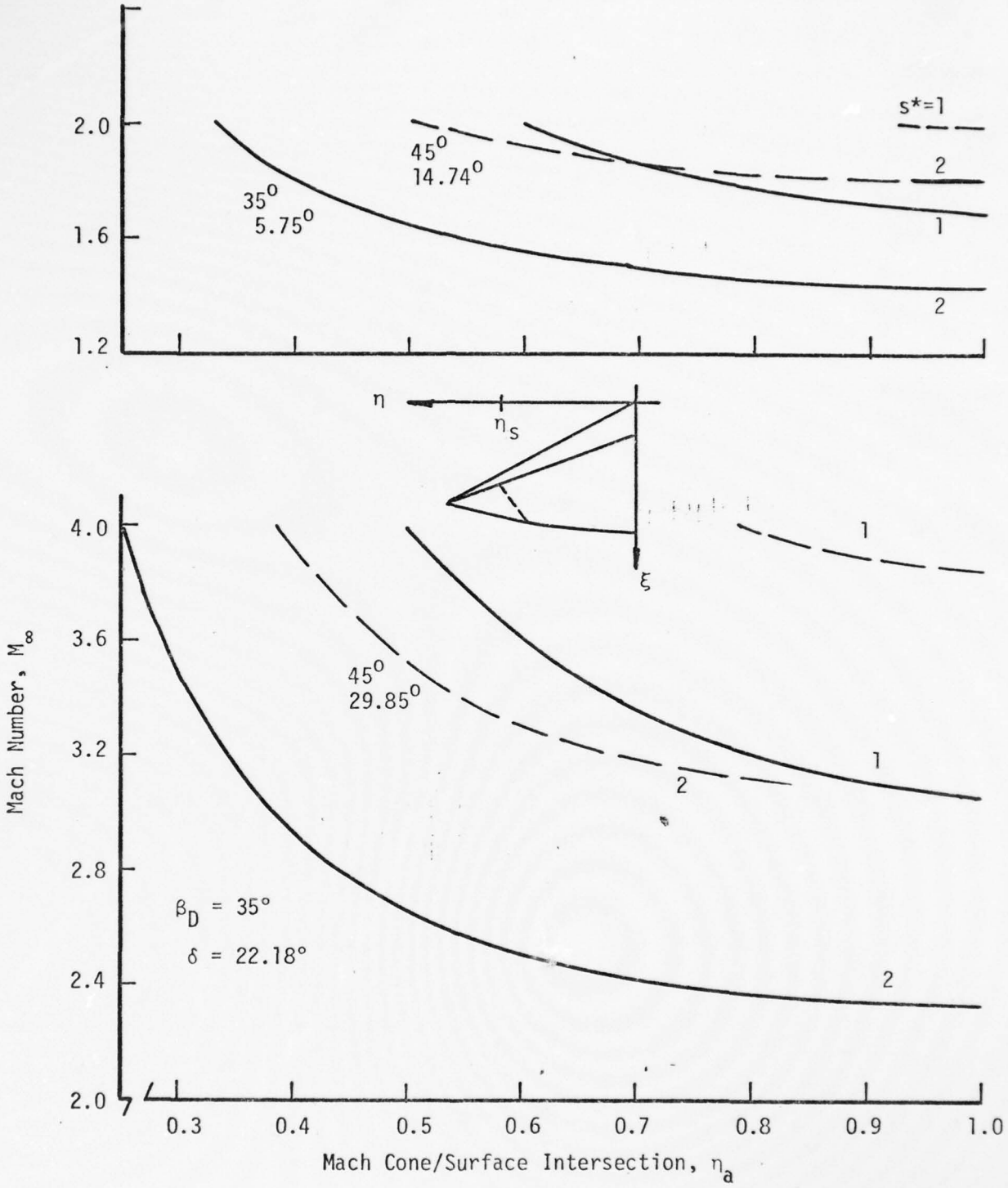


Figure 11. MACH CONE INTERSECTION WITH SURFACE, ATTACHED SHOCK

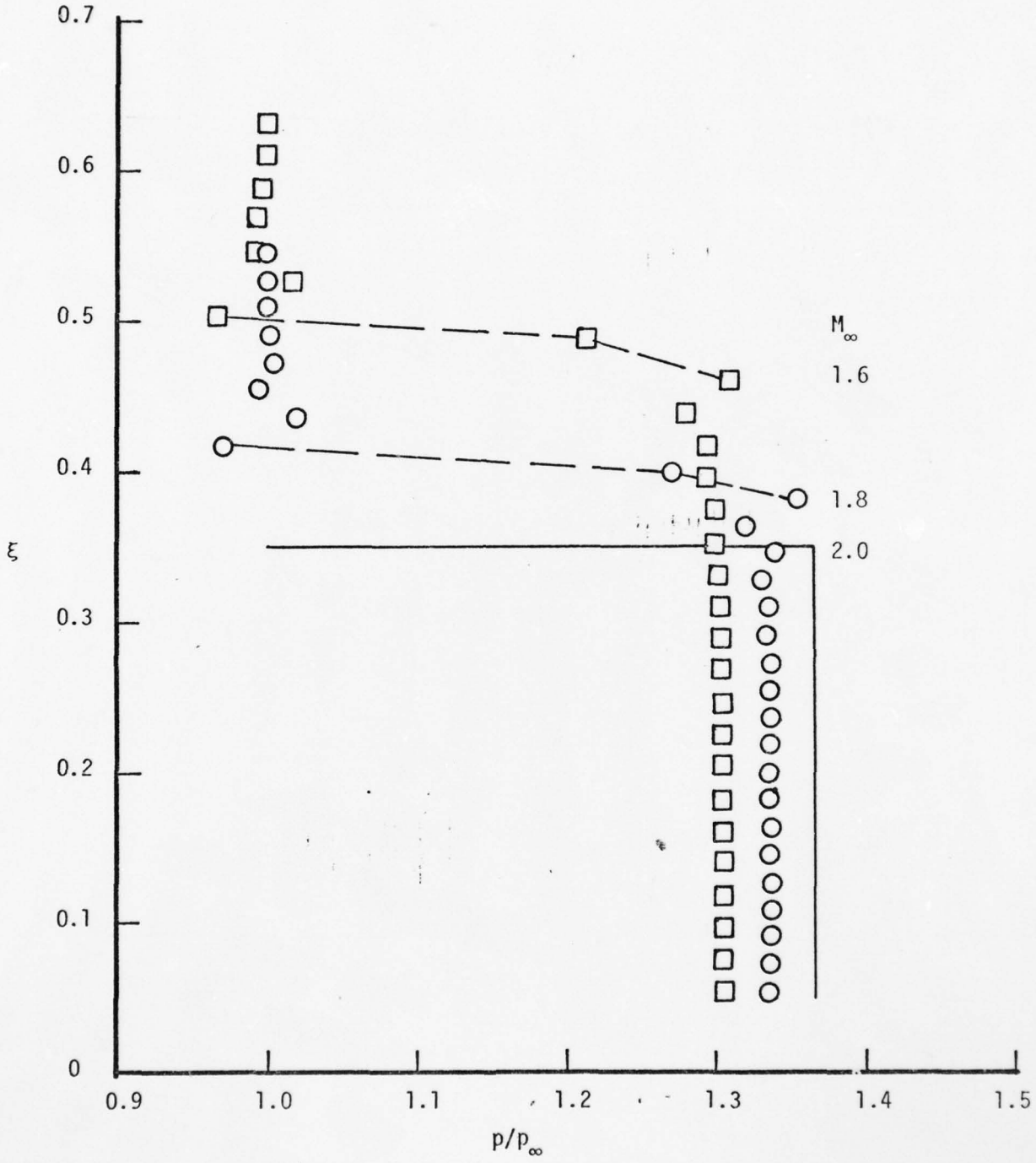


Figure 12. PRESSURE DISTRIBUTIONS FROM SURFACE TO FREE STREAM ALONG SYMMETRY PLANE: $s^*=2$, $\beta_D=35^o$
(a) $\delta = 5.75^o$, $M_\infty = 1.6, 1.8, 2.0$

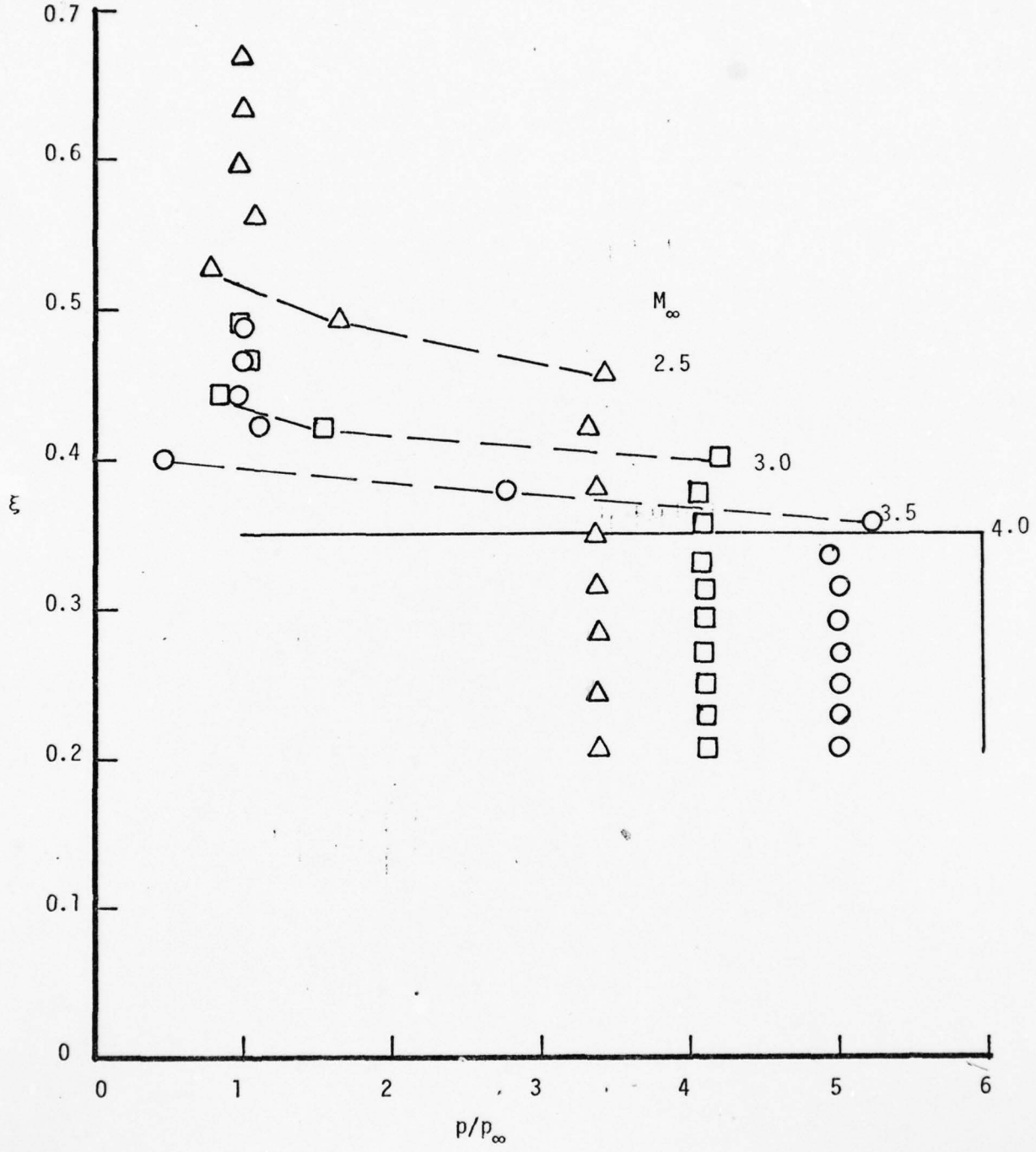


Figure 12 (concluded) (b) $\delta = 22.18^{o}$, M = 2.5, 3.0, 3.5, 4.0

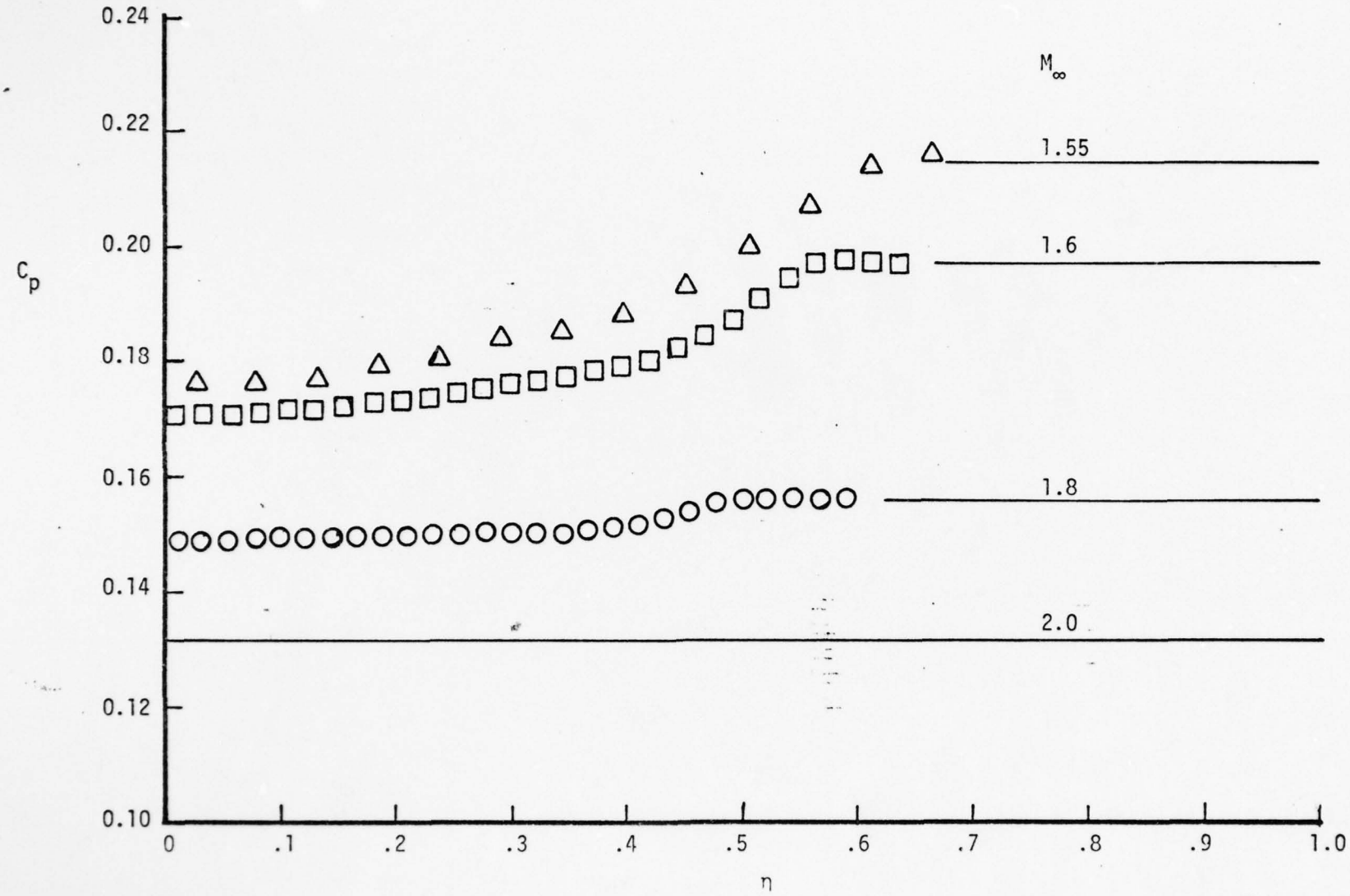


Figure 13. SURFACE PRESSURE COEFFICIENT DISTRIBUTION; $s^*=2$, $\beta_D=35^o$
(a) $\delta = 5.75^o$, $M_\infty = 1.55, 1.6, 1.8, 2.0$

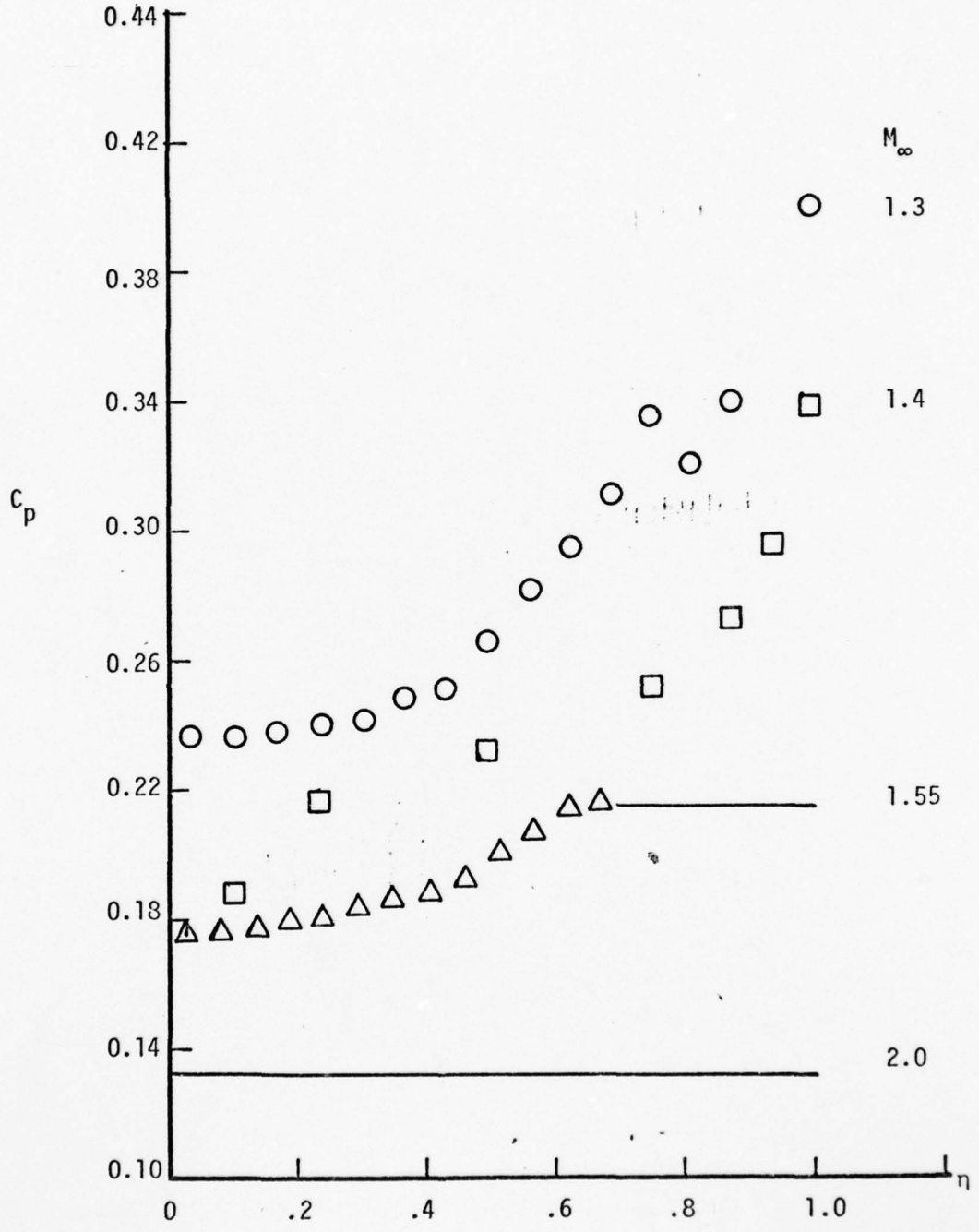


Figure 13 (continued). (b) $\delta = 5.75^{o}$, M_∞ = 1.3, 1.4, 1.55, 2.0

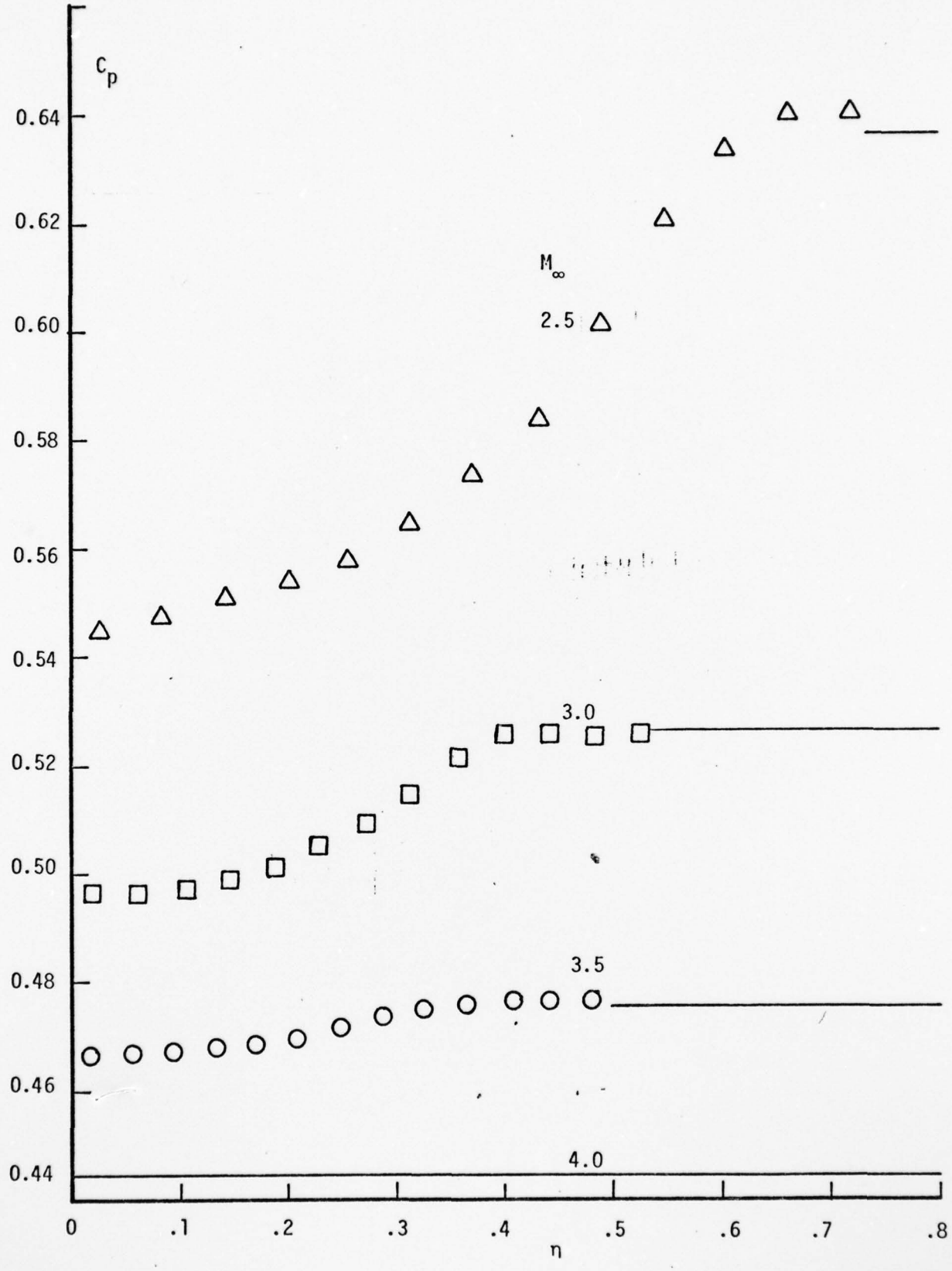


Figure 13 (concluded) (c) $\delta = 22.18^{o}$, $M_\infty = 2.5, 3.0, 3.5, 4.0$

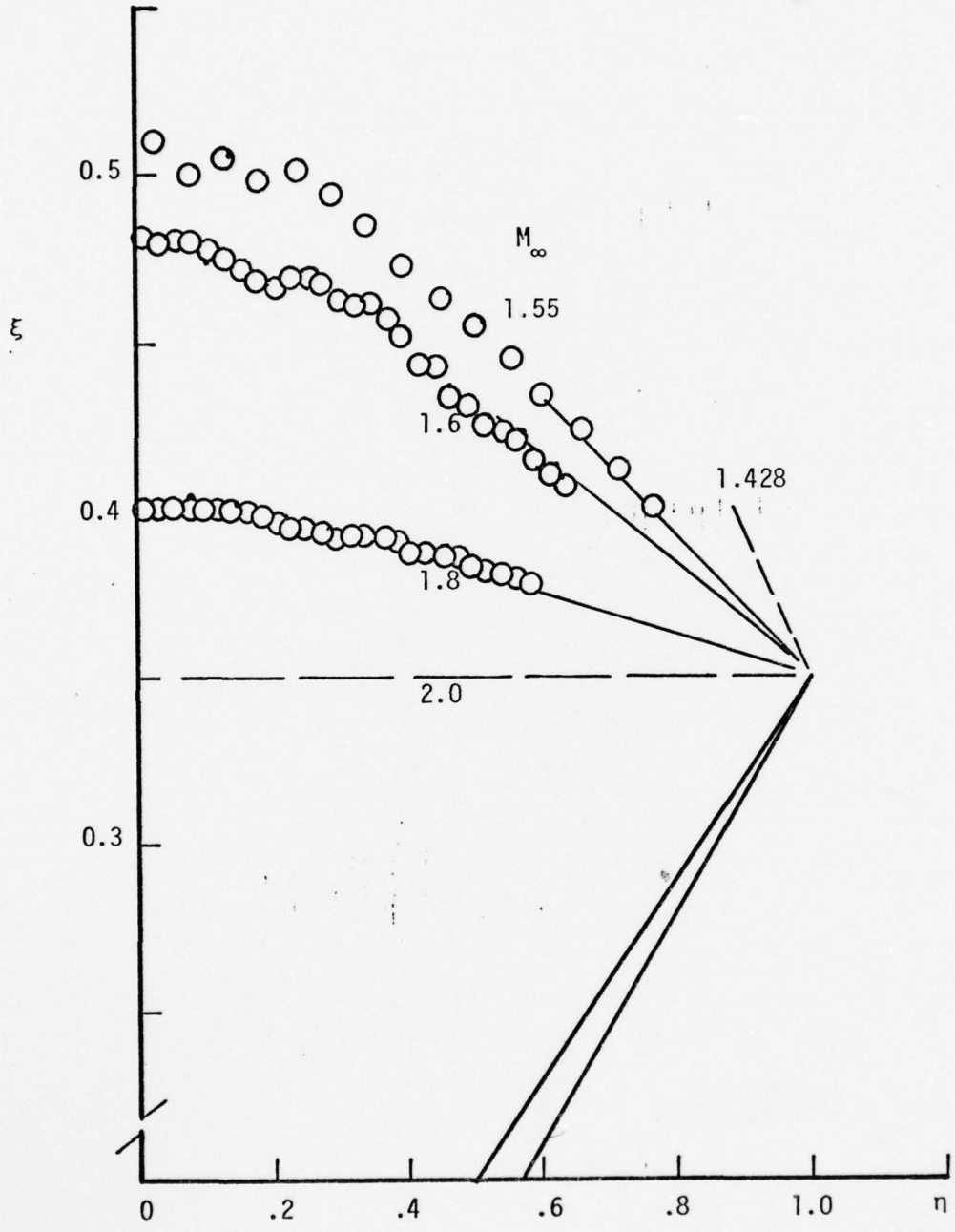


Figure 14. SHOCK TRACES; $s^*=2$, $\beta_D = 35^o$
(a) $\delta = 5.75^o$, $M_\infty = 1.55, 1.6, 1.8, 2.0$

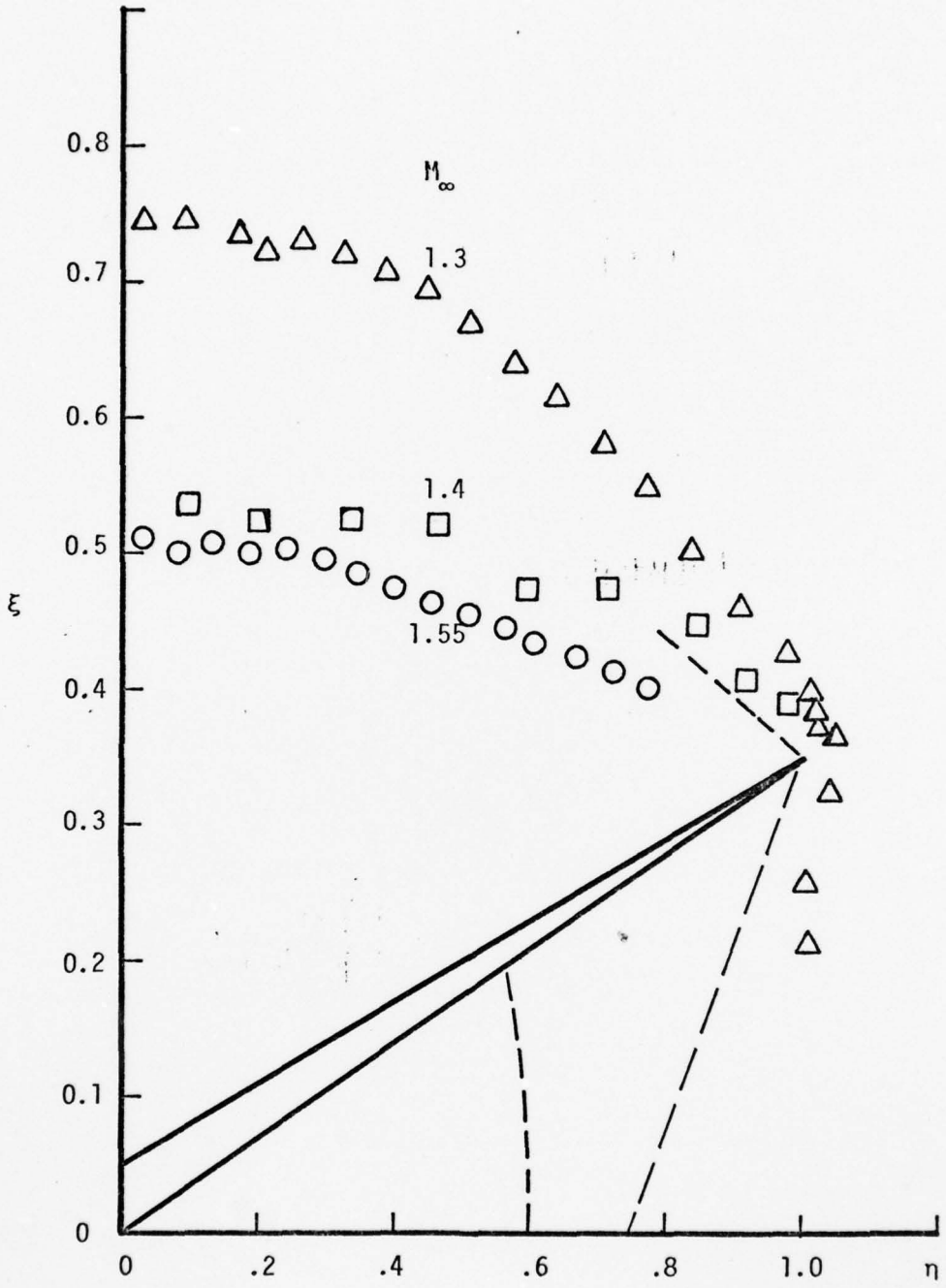


Figure 14 (continued) (b) $\delta = 5.75^{o}$, $M_{\infty} = 1.3, 1.4, 1.55$

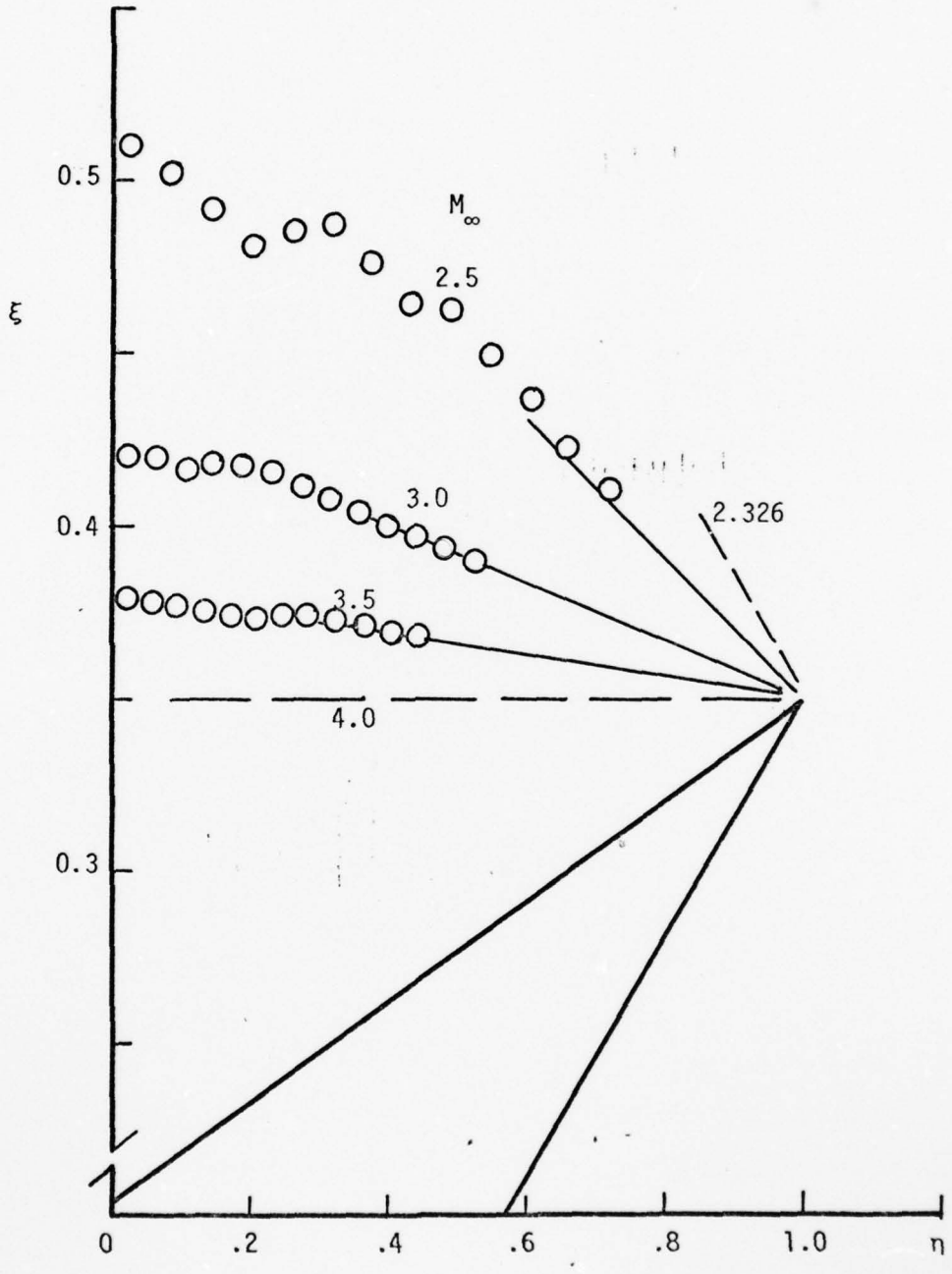


Figure 14 (concluded) (c) $\delta = 22.18^{o}$, $M_\infty = 2.5, 3.0, 3.5, 4.0$

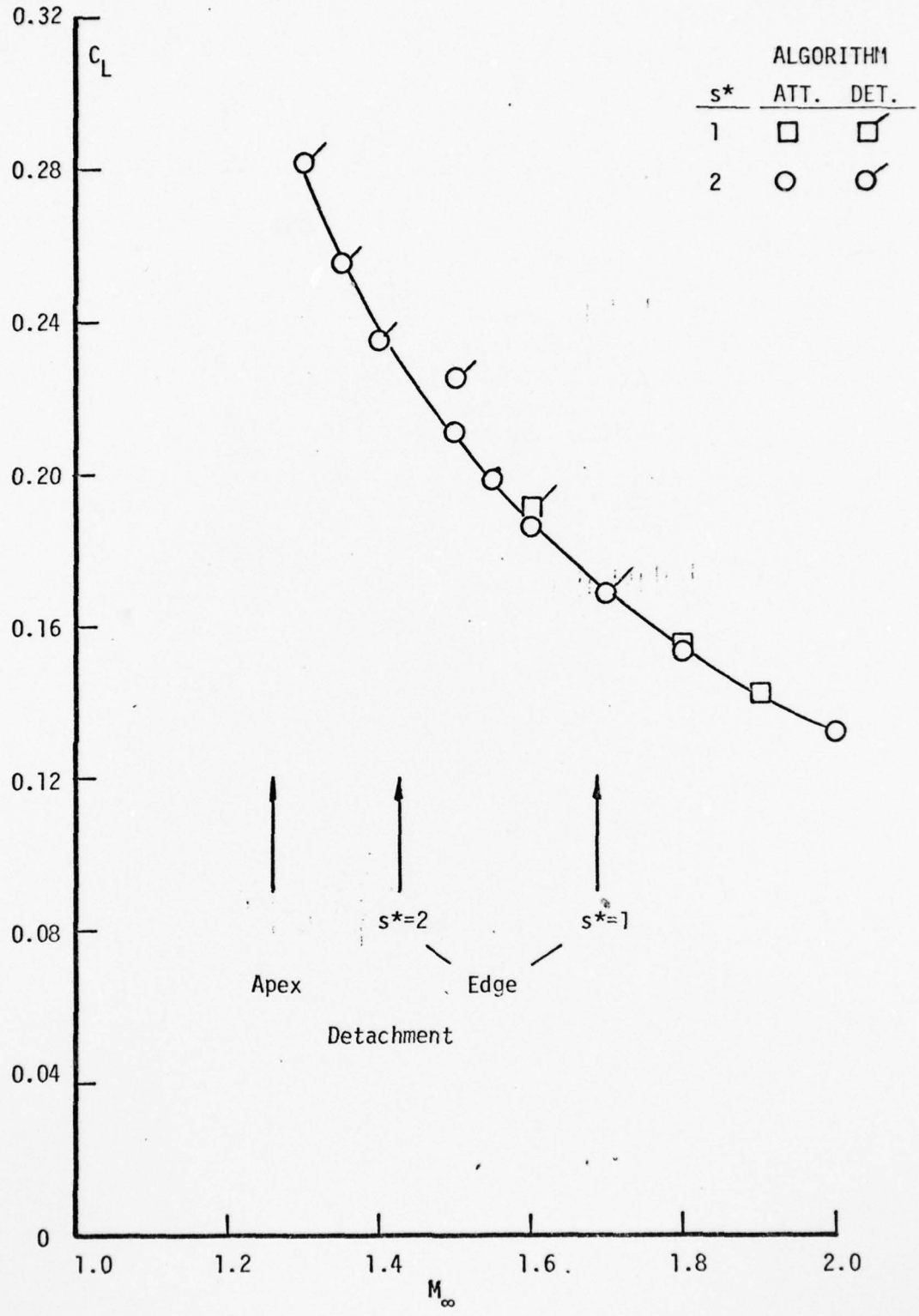


Figure 15. LIFT COEFFICIENT VARIATION WITH MACH NUMBER; $\delta = 5.75^{o}$, $\beta_D = 35^{o}$

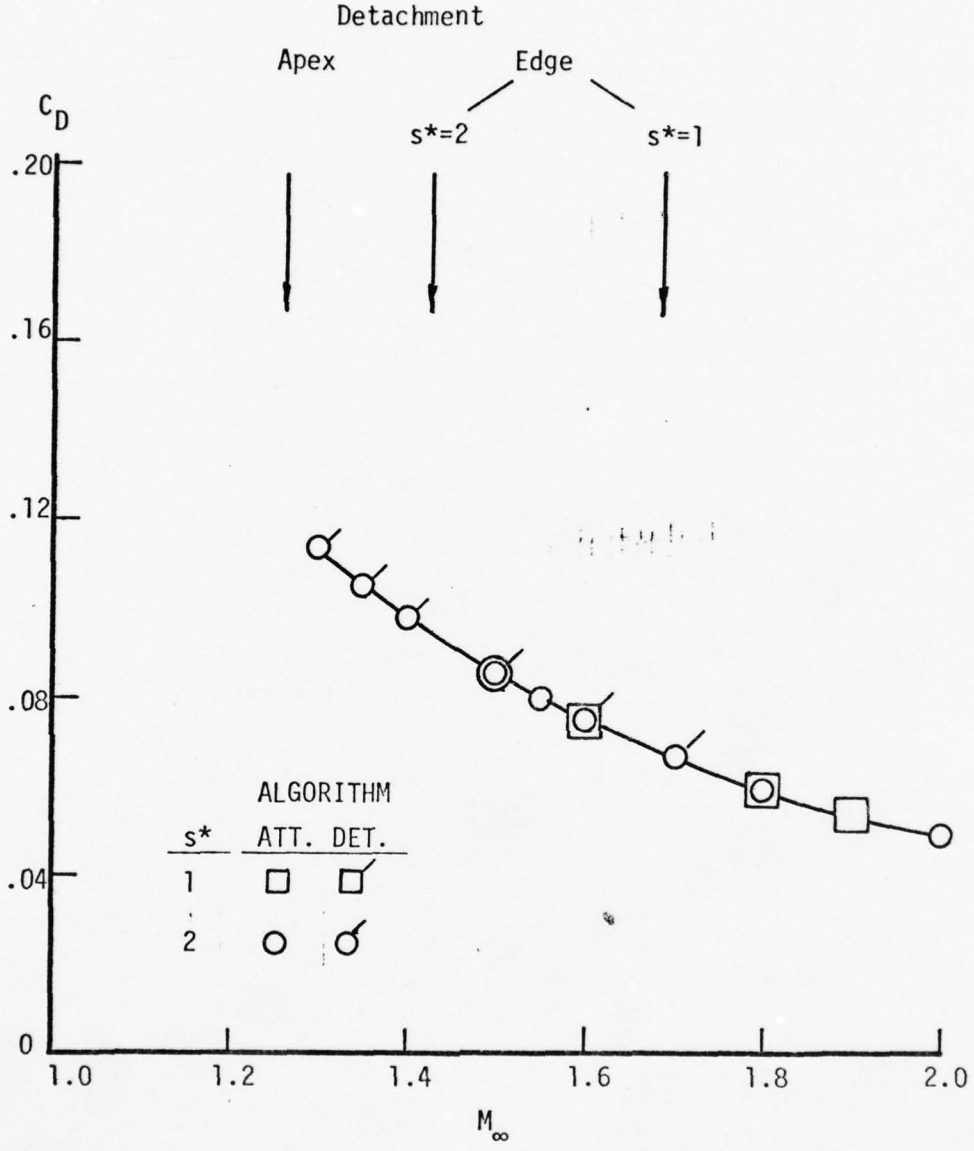


Figure 16. DRAG COEFFICIENT VARIATION WITH MACH NUMBER;
$\delta = 5.75^o$, $\beta_D = 35^o$

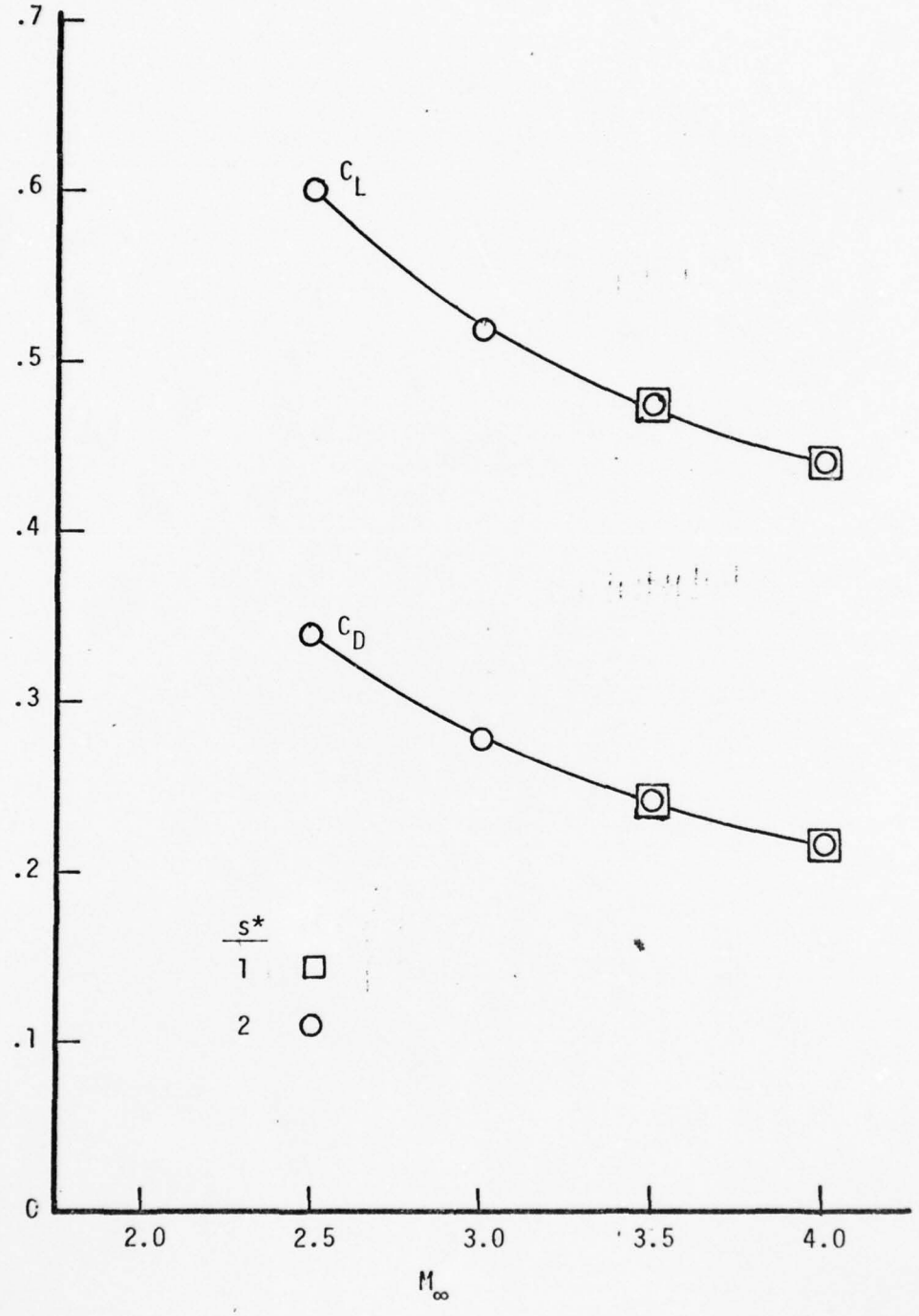


Figure 17. LIFT AND DRAG COEFFICIENT VARIATIONS WITH MACH NUMBER; $\delta = 22.18^{o}$, $\beta_D = 35^{o}$

REFERENCES

1. Nonweiler, T., Delta Wings of Shapes Amenable to Exact Shock Wave Theory, J. Roy. Aero. Soc., 67, 39-40 (1963).
2. Gonor, A.L., Conical Bodies of Minimum Drag in Hypersonic Gas Flow, PMM - J. Appl. Math & Mech., 28, 471-475, (1964).
3. Maikapar, G.I., On the Wave Drag of Non-Axisymmetric Bodies at Supersonic Speeds, PMM - J. Appl. Math & Mech., 23, 528-531, (1959).
4. Kuchemann, D., Hypersonic Aircraft and Their Aerodynamic Problems, Progress in Aeronautical Science, 6, 271-353, Pergamon Press, N.Y. (1965).
5. Seddon, J. and Spence, A., The Use of Known Flow Fields as an Approach to the Design of High Speed Aircraft, Hypersonic Boundary Layers and Flow Fields, AGARD C.P. 30, (1968).
6. Hui, W.H., The Caret Wing at Certain Off-Design Conditions, The Aero Quarterly, Roy. Aero. Soc., 23, 263-275, (1972).
7. Ganzer, A., An Exact Method of Calculation for the Inviscid Flow Field at the Bottom of Caret Wings, Zeitschrift fur Flugwissenschaften, 4, 109-116, (1975).
8. Gonor, A.L., Theory of Hypersonic Flow About A Wing, Progress in Aerospace Sciences, 14, 109-175, (1973).
9. Lan, T.H., Numerical Flow Field Analysis for Off Design Caret Wings Using Sharp Shock Propagation, SM Thesis, M.I.T., (1978).
10. Squire, L.C., Calculated Pressure Distributions and Shock Shapes on Thick Conical Wings at High Supersonic Speeds, The Aero Quarterly, Roy. Aero. Soc., 18, 185-206, (1967).

11. Greenwood, G.H., Free Flight Measurements of Pressure and Heat Transfer on a Blunt Leading Edge Caret Wing at Design and Off-Design Mach Numbers of 0.9 - 3.36, R & M No. 3679 (1970).

12. MacCormack, R.W., The Effect of Viscosity in Hypervelocity Impact Cratering, AIAA Paper 69-354, (1969).